

# EUROPEAN PATENT OFFICE

## Patent Abstracts of Japan

PUBLICATION NUMBER : 09231402  
PUBLICATION DATE : 05-09-97

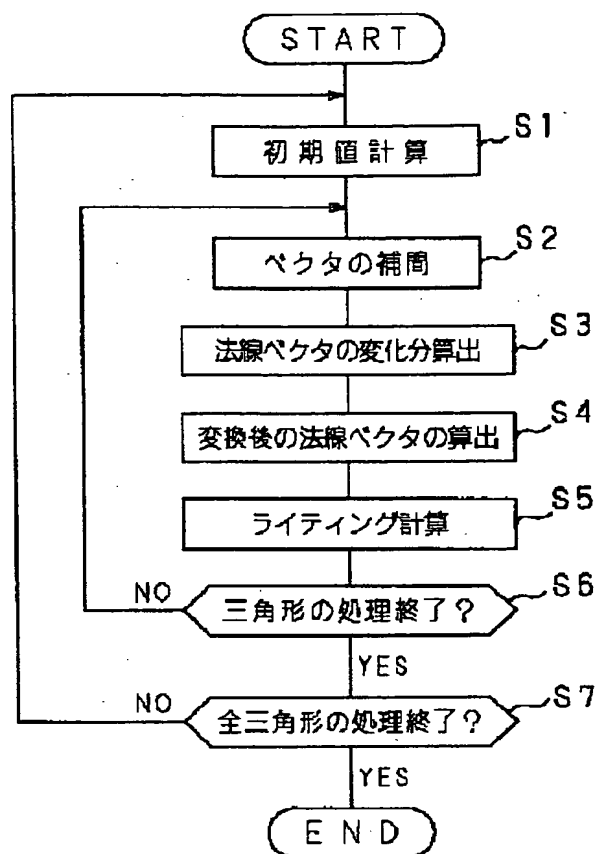
APPLICATION DATE : 27-02-96  
APPLICATION NUMBER : 08040170

APPLICANT : SONY CORP;

INVENTOR : YAMAGUCHI YUJI;

INT.CL. : G06T 15/50

TITLE : BUMP MAPPING METHOD AND  
PICTURE GENERATING DEVICE



**ABSTRACT :** PROBLEM TO BE SOLVED: To provide a bump mapping method capable of quickly executing bump mapping with a small hardware.

**SOLUTION:** A patch defined by two variables (u), (v) is approximated by a set of polygons and partial differential for the variables (u), (v) of the patch is calculated (S1). Then, a normal vector for a pixel to be plotted is calculated by the use of the partial differential (S2). Then, the variation of the normal vector is calculated by the use of a texture function (S3). Then, a transformed normal vector for calculating the color value of the pixel to be plotted is calculated by the use of the normal vector and the variation of the normal vector (S4).

COPYRIGHT: (C)1997,JPO

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-231402

(43) 公開日 平成9年(1997)9月5日

(51) Int. Cl.<sup>4</sup>  
G 0 6 T 15/50

識別記号 片内整理番号

P I  
G 0 6 F 15/72

技術表示箇所

4 6 5

審査請求 未請求 請求項の数12 O L (全 18 頁)

(21) 出願番号 特願平3-40170

(22) 出願日 平成8年(1996)2月27日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 山口 裕司

東京都品川区東五反田1-14-10 株式会社

社ソニー木原研究所内

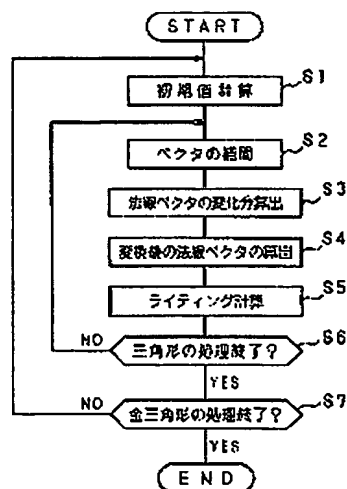
(74) 代理人 弁護士 小池 晃 (外2名)

(54) 【発明の名称】 バンプマッピング方法及び画像生成装置

(57) 【要約】

【課題】 小規模のハードウェアでバンプマッピングを高速に実行することができるバンプマッピング方法を提供する。

【解決手段】 先ず、2つの変数 $u$ 及び $v$ を用いて定義されるパッチを多角形の集合で近似して、上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出する(S1)。次に、上記偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する(S2)。次に、テクスチャ関数を用いて上記法線ベクトルの変化分を算出する(S3)。そして、上記法線ベクトルと上記法線ベクトルの変化分を用いて、対象描画ピクセルのカラー値を算出するための変換後法線ベクトルを算出する(S4)。



本発明に係るバンプマッピング方法

(2)

特開平9-231402

1

2

## 【特許請求の範囲】

【請求項1】 2つの変数 $u$ 及び $v$ を用いて定義されるパッチを描画する際に、対象描画ピクセルに対する法線ベクトルをテクスチャ関数により変換し、変換して得られた変換後法線ベクトルを用いて対象描画ピクセルのカラー値を算出するバンプマッピング方法であって、

上記パッチを多角形の集合で近似し、

上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出し、

上記偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出し、

テクスチャ関数を用いて上記法線ベクトルの変化分を算出し、

上記法線ベクトルと上記法線ベクトルの変化分を用いて上記変換後法線ベクトルを算出することを特徴とするバンプマッピング方法。

【請求項2】 上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数 $u$ 及び $v$ の値と座標値から上記偏微分を算出することを特徴とする請求項1記載のバンプマッピング方法。

【請求項3】 上記法線ベクトルを多角形の各頂点における法線ベクトルから算出することを特徴とする請求項1記載のバンプマッピング方法。

【請求項4】 上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出することを特徴とする請求項1記載のバンプマッピング方法。

【請求項5】 上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出することを特徴とする請求項1記載のバンプマッピング方法。

【請求項6】 ライティング計算における拡散反射に関わる項を予め算出し、その算出結果を拡散反射テーブルに格納し、

ライティング計算における鏡面反射に関わる項を予め算出し、その算出結果を鏡面反射テーブルに格納し、

上記変換後法線ベクトルで上記拡散反射テーブルを参照し、

視線に対する反射ベクトルで上記鏡面反射テーブルを参照して上記カラー値を算出することを特徴とする請求項1記載のバンプマッピング方法。

【請求項7】 2つの変数 $u$ 及び $v$ を用いて定義されるパッチを描画する際に、対象描画ピクセルに対する法線ベクトルをテクスチャ関数により変換し、変換して得られた変換後法線ベクトルを用いて対象描画ピクセルのカラー値を算出するバンプマッピング方法を採用した画像生成装置であって、

上記パッチを多角形の集合で近似して上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出する偏微分算出手段

と、

上記偏微分算出手段で得られた偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する法線算出手段と、

上記法線算出手段で得られた法線ベクトルの変化分をテクスチャ関数を用いて算出する変化分算出手段と、

上記法線算出手段で得られた法線ベクトルと上記変化分算出手段で得られた法線ベクトルの変化分を用いて上記変換後法線ベクトルを算出する変換後法線算出手段とを備えることを特徴とする画像生成装置。

【請求項8】 上記偏微分算出手段は、上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数 $u$ 及び $v$ の値と座標値から上記偏微分を算出することを特徴とする請求項7記載の画像生成装置。

【請求項9】 上記法線算出手段は、上記法線ベクトルを多角形の各頂点における法線ベクトルから算出することを特徴とする請求項7記載の画像生成装置。

【請求項10】 上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出することを特徴とする請求項7記載の画像生成装置。

【請求項11】 上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出することを特徴とする請求項7記載の画像生成装置。

【請求項12】 予め算出されたライティング計算における拡散反射に関わる項が格納される拡散反射テーブルと、

予め算出されたライティング計算における鏡面反射に関わる項が格納される鏡面反射テーブルと、

上記変換後法線算出手段で得られた変換後法線ベクトルで上記拡散反射テーブルを参照し、視線に対する反射ベクトルで上記鏡面反射テーブルを参照して上記カラー値を算出するカラー値算出手段とを備えることを特徴とする請求項7記載の画像生成装置。

## 【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータを用いた映像機器であるコンピュータグラフィックスシステム、特殊効果装置、ビデオゲーム機等において、物体上の小さな凹凸をシェーディングによる陰影づけにより表現するためのバンプマッピング方法、及び上記バンプマッピング方法を採用した画像生成装置に関するものである。

【0002】

【従来の技術】例えば、コンピュータグラフィックスシステムは、計算機とグラフィックス周辺装置により、画像や映像を作成して表示するシステムであり、機械、電気、建築等における設計支援のためのCADシステム、

(3)

特開平9-231402

3

4

化学、航空、制御等における反応や応答のシミュレーション、教育、芸術、及びビデオゲーム等、多くの分野において活用されている。

【0003】コンピュータグラフィックスシステムとして、主として計算機の数値計算能力を活用して立体的な画像を作成する画像生成装置を備えたシステム（以下、3次元グラフィックスシステムと言う。）があり、この3次元グラフィックスシステムは、計算機中にある立体モデルを、回転、移動、拡大／縮小という座標変換を頻繁に行って画面表示するシステムであり、2次元グラフィックスシステムに比べ、座標変換、透視変換、陰影処理、及び隠線／隠面消去処理等、高度な技術を必要とするシステムである。

【0004】上述のような3次元グラフィックスシステムの応用分野は、近年において、急激に拡大しており、CADのみならず、映像制作やビデオゲーム等においても広く利用されている。

【0005】このため、3次元グラフィックスシステムが備える画像生成装置には、単に物体の3次元の形状を2次元の画面上に表示するだけではなく、よりリアリ

ティの高い画像を生成する能力が要求されている。  
【0006】そこで、グラフィックスの表現力を向上させた画像を得るために、上記画像生成装置では、テクスチャマッピングと呼ばれる手法が採用されている。このテクスチャマッピングとは、テクスチャソース画像として別に用意された2次元画像（絵柄）、すなわちテクスチャパターンを物体を構成する三角形や四角形等の基本的な単位図形（ポリゴン）の表面に張り付ける技術であり、3次元グラフィックスシステムの中には、テクスチャマッピングをハードウェアで高速に実行する画像生成装置を備えるものもある。

【0007】しかし、上述したようなテクスチャマッピングは、色に関する表現力を大幅に向上させることはできるが、テクスチャマッピングにより得られた画像は、物体の表面が滑らかすぎてリアリティに欠けるという問題があった。

【0008】そこで、この問題を解決するために、テクスチャマッピングを発展させたバンプマッピングと呼ばれる手法を採用した画像生成装置がある。このバンプマッピングとは、法線ベクトルをマッピングして、物体上の小さな凹凸をシェーディングによる陰影づけによって表現することにより、物体表面上の不均質さや凹凸を疑似的に表現するという技術である。

【0009】図5は、バンプマッピングの処理を示すフローチャートである。以下、上記図5を用いて、バンプマッピングの処理について説明する。

【0010】例えば、物体が2つの変数  $u$ 、 $v$  の多項式で表される曲線でモデリングされている場合、まず、ステップS101では、フレームメモリ上に描画するピクセル（以下、描画ピクセルと言う。）に対応する物体上

の点Pにおける位置ベクトルPを、

【0011】

【数1】

$$P = (X(u, v), Y(u, v), Z(u, v)) \quad \dots (1)$$

【0012】なる式（1）により算出する。

【0013】そして、式（1）により得られた位置ベクトルPの“ $u$ ”及び“ $v$ ”に対する偏微分  $P_u$ 、 $P_v$  を、

【0014】

【数2】

$$P_u = (\partial X / \partial u, \partial Y / \partial u, \partial Z / \partial u) \quad \dots (2)$$

【0015】

【数3】

$$P_v = (\partial X / \partial v, \partial Y / \partial v, \partial Z / \partial v) \quad \dots (3)$$

【0016】なる式（2）及び（3）により算出する。

【0017】ここで、式（2）及び（3）により得られた偏微分  $P_u$ 、 $P_v$  は、物体上の点Pにおける接平面上に存在する。

【0018】そこで、次のステップS102では、点Pにおける法線ベクトルNを、式（1）により得られた偏微分  $P_u$  及び  $P_v$  の外積により求める。すなわち、ステップS102では、法線ベクトルNを、

【0019】

【数4】

$$N = P_u \times P_v \quad \dots (4)$$

【0020】なる式（4）により算出する。

【0021】次に、ステップS103では、まず、物体の面上の凹凸（以下、バンプとも言う。）の高さを与える関数を“ $F(u, v)$ ”とし、“ $F$ ”の“ $u$ ”及び“ $v$ ”に対する偏微分  $F_u$  及び  $F_v$  をテクスチャとする。

【0022】そして、物体上の点Pにおける“ $u$ ”及び“ $v$ ”の各値から対応するテクスチャの値を参照し、ステップS101で得られた偏微分  $P_u$ 、 $P_v$  と、ステップS102で得られた法線ベクトルNとを用いて、法線ベクトルNの変化分Dを、

【0023】

【数5】

$$D = \frac{F_u * (N \times P_v) - F_v * (N \times P_u)}{|N|} \quad \dots (5)$$

【0024】なる式（5）により求める。

【0025】尚、式（5）において、法線ベクトルNが、

$$N = (N_x, N_y, N_z)$$

(4)

特開平9-231402

5

であるとき、 $-|N|$  は、

【0026】

【数6】

$$|N| = \sqrt{N_x^2 + N_y^2 + N_z^2} \quad \dots (6)$$

【0027】なる式(6)で与えられる。

【0028】次に、ステップS104では、ステップS102で得られた法線ベクトルNと、ステップS103で得られた法線ベクトルNの変化分Dとを、

【0029】

【数7】

$$N^* = N + D \quad \dots (7)$$

【0030】なる式(7)に示すように、加算することにより、バンプが付いた面の法線ベクトル $N^*$ を求める。

10

\* 【0031】次に、ステップS105では、ステップS104で得られた法線ベクトル $N^*$ を、

【0032】

【数8】

$$N_n = \frac{N^*}{|N^*|} \quad \dots (8)$$

【0033】なる式(8)により正規化することにより、正規化法線ベクトル $N_n$ を求める。

【0034】尚、式(8)において、法線ベクトル $N^*$ が、

$$N^* = (N^*_x, N^*_y, N^*_z)$$

であるとき、 $-|N^*|$  は、

【0035】

【数9】

$$|N^*| = \sqrt{N^*_x^2 + N^*_y^2 + N^*_z^2} \quad \dots (9)$$

【0036】なる式(9)で与えられる。

【0037】次に、ステップS106では、ステップS105で得られた正規化法線ベクトル $N_n$ を用いて、ライティング計算を行い、対象描画ピクセルの色を決定する。

【0038】すなわち、ステップS106では、ライティング計算を、環境光(Ambient Light)のカラーI

a. 照明のカラーI p、ポジションライト(Position Light)の減衰率f a t t、環境反射係数(Ambient-reflection Coefficient) k a、拡散反射係数(Diffuse-reflection Coefficient) k d、鏡面反射係数(Specular-reflection Coefficient) k s、物体の拡散カラー(Diffuse Color) O d、物体のスペキュラーカラー(Specular Color) O s、法線ベクトルN、視線ベクトルV、ライトに対する反射ベクトルR、ライトベクトルL、及びスペキュラーのべき指数(Specular-reflection Exponent) nを持って、

\* flection Coefficient) k d、鏡面反射係数(Specular-reflection Coefficient) k s、物体の拡散カラー(Diffuse Color) O d、物体のスペキュラーカラー(Specular Color) O s、法線ベクトルN、視線ベクトルV、ライトに対する反射ベクトルR、ライトベクトルL、及びスペキュラーのべき指数(Specular-reflection Exponent) nを持って、

【0039】

【数10】

$$I \lambda = I a \lambda k a O d \lambda + \sum f a t t I p \lambda [k d O d \lambda (N \cdot L) + k s O s \lambda (R \cdot V)^n] \quad \dots (10)$$

【0040】なる式(10)により行うことにより、対象描画ピクセルの色を決定する。

【0041】ところで、式(10)において、 $\lambda$  は、各カラーコンポーネント毎に値を持つことを示す。

【0042】また、ライトに対する反射ベクトルRは、

【0043】

【数11】

$$R = 2 * N * (N \cdot L) - L \quad \dots (11)$$

【0044】なる式(11)で与えられる。

【0045】また、式(10)及び式(11)において、法線ベクトルN、視線ベクトルV、ライトに対する反射ベクトルR、及びライトベクトルLは、各々、正規化されているものとする。

【0046】尚、上述したライティング計算については、“Computer Graphics principles and practice (Foley, Van Dam他著)”や“ADDISON WESLEY”の文献等に記載されているため、その詳細な説明は省略する。

【0047】次に、ステップS107では、ステップS101～ステップS106の各処理が全ての描画ピクセルに対して行われたか否かを判断する。そして、全ての描画ピクセルへの処理が終了するまで、ステップS101～ステップS106の各処理を描画ピクセル毎に行う。

40

【0048】ところで、上記図5に示したようなバンプマッピングの処理により得られた画像において、物体表面の疑似的な凹凸の見え方は、バンプマッピングを施される物体の大きさに依存する。すなわち、上記バンプマッピングの処理では、バンプマッピングを施される物体を大きくしていくと、物体表面の凹凸が平坦化されていく。

【0049】しかし、3次元グラフィックスシステムの応用分野によっては、物体表面の疑似的な凹凸の見え方が物体の大きさに依存すると都合が悪い場合がある。

50

【0050】そこで、このような場合には、ステップS

(5)

特開平9-231402

103において、式(5)を、  
【0051】

$$D = \sqrt{Fu^2 + Fv^2} \cdot |N| \cdot \frac{Fu \cdot (N \times Pv) - Fv \cdot (N \times Pu)}{|Fu \cdot (N \times Pv) - Fv \cdot (N \times Pu)|} \quad \dots (12)$$

【0052】なる式(12)に置き換えて、法線ベクトルNの変化分Dを求める。これにより、物体の大きさに依存しないバンプマッピングを実現することができる。

【0053】尚、上述したバンプマッピングについては、“SIMULATION OF WRINKLED SURFACES (James F. Blinn著)”や“Proceed of SIGGRAPH '78”の論文等に記載されているため、その詳細な説明は省略する。

【0054】

【発明が解決しようとする課題】しかし、上述したような従来のバンプマッピング方法では、ステップS101で述べたように、描画ピクセルに対応する位置ベクトルを描画ピクセル毎に求めると共に、上記位置ベクトルの偏微分を描画ピクセル毎に求める必要があった。このため、上記バンプマッピング方法では、バンプマッピングの処理を行う際の計算量が大きくなってしまっていた。

【0055】また、上記バンプマッピング方法では、ステップS102で述べたように、ステップS101で得られた位置ベクトルの偏微分を用いて、法線ベクトルを描画ピクセル毎に求める必要があった。このため、上記バンプマッピング方法では、バンプマッピングの処理を行う際の計算量が大きくなってしまっていた。

【0056】また、上記バンプマッピング方法では、ステップS103の式(5)又は式(12)に示したように、描画ピクセル毎の法線ベクトルの変化分を算出する際、ルートの計算、除算、及びベクトルの外積等の計算が必要であった。このため、上記バンプマッピング方法では、バンプマッピングの処理を行う際の計算量が大きくなってしまっていた。

【0057】また、上記バンプマッピング方法では、ステップS106で述べたように、描画ピクセル毎のライティング計算を行う際、法線ベクトル、視線ベクトル、ライトベクトル、及びライトに対する反射ベクトルの正規化が必要であり、且つ式(10)に示したように、ベキ乗の計算も必要であった。このため、上記バンプマッピング方法では、バンプマッピングの処理を行う際の計算量が大きくなってしまっていた。

【0058】したがって、上述のようなバンプマッピングをハードウェア化しようとした場合、ハードウェア規模が非常に大きくなってしまいうため、バンプマッピングをハードウェアで実現することが困難であった。すなわち、バンプマッピングを採用した従来の画像生成装置は、バンプマッピングをハードウェアで実行することができなかった。

【0059】そこで、本発明は、上述の如き従来の実情

\*【数12】

10

20

30

40

50

に鑑みてなされたものであり、次のような目的を有するものである。

【0060】即ち、本発明の目的は、小規模のハードウェアでバンプマッピングを高速に実行することができるバンプマッピング方法を提供することにある。

【0061】また、本発明の目的は、小規模のハードウェアでバンプマッピングを高速に実行する画像生成装置を提供することにある。

【0062】

【課題を解決するための手段】上述の課題を解決するために、本発明に係るバンプマッピング方法は、2つの変数u及びvを用いて定義されるパッチを描画する際に、対象描画ピクセルに対する法線ベクトルをテクスチャ関数により変換し、変換して得られた変換後法線ベクトルを用いて対象描画ピクセルのカラー値を算出するバンプマッピング方法であって、先ず、上記パッチを多角形の集合で近似する。次に、上記パッチの変数u及びv各々に対する偏微分を算出する。次に、上記偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する。次に、テクスチャ関数を用いて上記法線ベクトルの変化分を算出する。そして、上記法線ベクトルと上記法線ベクトルの変化分を用いて上記変換後法線ベクトルを算出することを特徴とする。

【0063】また、本発明に係るバンプマッピング方法は、上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数u及びvの値と座標値から上記偏微分を算出することを特徴とする。

【0064】また、本発明に係るバンプマッピング方法は、上記法線ベクトルを多角形の各頂点における法線ベクトルから算出することを特徴とする。

【0065】また、本発明に係るバンプマッピング方法は、上記変数uに対する偏微分と上記変数vに対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出することを特徴とする。

【0066】また、本発明に係るバンプマッピング方法は、上記変数uに対する偏微分と上記変数vに対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出することを特徴とする。

【0067】また、本発明に係るバンプマッピング方法は、ライティング計算における拡散反射に関わる項を予め算出し、その算出結果を拡散反射テーブルに格納する。また、ライティング計算における鏡面反射に関わる項を予め算出し、その算出結果を鏡面反射テーブルに格

(6)

特開平9-231402

9

10

納する。そして、上記変換後法線ベクトルで上記拡散反射テーブルを参照し、視線に対する反射ベクトルで上記鏡面反射テーブルを参照して上記カラー値を算出することを特徴とする。

【0068】上述の問題を解決するために、本発明に係る画像生成装置は、2つの変数 $u$ 及び $v$ を用いて定義されるパッチを描画する際に、対象描画ピクセルに対する法線ベクトルをテクスチャ関数により変換し、変換して得られた変換後法線ベクトルを用いて対象描画ピクセルのカラー値を算出するバンプマッピング方法を採用した画像生成装置であって、上記パッチを多角形の集合で近似して上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出する偏微分算出手段と、上記偏微分算出手段で得られた偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する法線算出手段と、上記法線算出手段で得られた法線ベクトルの変化分をテクスチャ関数を用いて算出する変化分算出手段と、上記法線算出手段で得られた法線ベクトルと上記変化分算出手段で得られた法線ベクトルの変化分を用いて上記変換後法線ベクトルを算出する変換後法線算出手段とを備えることを特徴とする。

【0069】また、本発明に係る画像生成装置は、上記偏微分算出手段は、上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数 $u$ 及び $v$ の値と座標値から上記偏微分を算出することを特徴とする。

【0070】また、本発明に係る画像生成装置は、上記法線算出手段は、上記法線ベクトルを多角形の各頂点における法線ベクトルから算出することを特徴とする。

【0071】また、本発明に係る画像生成装置は、上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出することを特徴とする。

【0072】また、本発明に係る画像生成装置は、上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出することを特徴とする。

【0073】また、本発明に係る画像生成装置は、予め算出されたライティング計算における拡散反射に関わる項が格納される拡散反射テーブルと、予め算出されたライティング計算における鏡面反射に関わる項が格納される鏡面反射テーブルと、上記変換後法線算出手段で得られた変換後法線ベクトルで上記拡散反射テーブルを参照し、視線に対する反射ベクトルで上記鏡面反射テーブルを参照して上記カラー値を算出するカラー値算出手段とを備えることを特徴とする。

【0074】

【発明の実施の形態】以下、発明の実施の形態について、図面を参照して詳細に説明する。

【0075】まず、本発明に係るバンプマッピング方法

について説明する。

【0076】例えば、2つの変数 $u$ 、 $v$ を用いて定義されるパッチに分割された物体を描画する際、先ず、上記パッチを多角形の集合で近似する。このとき、多角形の各頂点のモデリング座標系における座標値と法線ベクトル、及び $(u, v)$ 座標を頂点データとする。

【0077】次に、上記頂点データを多角形単位に座標変換することにより、プリミティブデータを生成する。そして、生成したプリミティブデータに対してバンプマッピングの処理を行う。

【0078】図1は、上記バンプマッピングの処理を示すフローチャートであり、上記バンプマッピングの処理は、本発明に係るバンプマッピング方法を適用したものである。

【0079】以下、上記図1を用いて、上記バンプマッピングの処理について、具体的に説明する。

【0080】例えば、三角形のプリミティブデータに対してバンプマッピングの処理を行う場合、先ず、ステップS1では、上記プリミティブデータを用いて、ステップS2におけるベクトルの補間処理、及びステップS3における法線ベクトルの変化分の算出処理に必要なパラメータを求める。

【0081】ステップS1の処理について具体的に説明すると、まず、三角形の各頂点のワールド座標系における座標 $P1, P2, P3$ が、

【0082】

【数13】

$$P1 = (X1, Y1, Z1) \quad \dots (13)$$

【0083】

【数14】

$$P2 = (X2, Y2, Z2) \quad \dots (14)$$

【0084】

【数15】

$$P3 = (X3, Y3, Z3) \quad \dots (15)$$

【0085】なる式(13)～(15)、ワールド座標系における法線ベクトル $N1, N2, N3$ が、

【0086】

【数16】

$$N1 = (Nx1, Ny1, Nz1) \quad \dots (16)$$

【0087】

【数17】

$$N2 = (Nx2, Ny2, Nz2) \quad \dots (17)$$

【0088】

【数18】

$$N3 = (Nx3, Ny3, Nz3) \quad \dots (18)$$

【0089】なる式(16)～(18)、 $(u, v)$ のパラメータ $T1, T2, T3$ が、

11	(7)	特開平9-231402
<p>【0090】 【数19】 T1 = (u1, v1) ... (19) 【0091】 【数20】 T2 = (u2, v2) ... (20) 【0092】 【数21】 T3 = (u3, v3) ... (21)</p> <p>【0093】なる式(19)～(21)、クリップ座標系における座標Pc1, Pc2, Pc3が、 【0094】 【数22】 Pc1 = (x1, y1, z1, w1) ... (22) 【0095】 【数23】 Pc2 = (x2, y2, z2, w2) ... (23) 【0096】 【数24】 Pc3 = (x3, y3, z3, w3) ... (24) 【0097】なる式(22)～(24)で各々与えられ、位置ベクトルの“u”及び“v”に対する偏微分Pu及びPvが、三角形内部で一定であると近似する。このとき、三角形の平面上の点(X, Y, Z)は、 【0098】 【数25】 X = ax*u + bx*v + cx ... (25) 【0099】 【数26】 Y = ay*u + by*v + cy ... (26) 【0100】 【数27】 Z = az*u + bz*v + cz ... (27) 【0101】なる式(25)～(27)で表される。 *</p>	<p>12</p> <p>*【0102】また、位置ベクトルの“u”及び“v”に対する偏微分Pu及びPvは、 【0103】 【数28】 Pu = (ax, ay, az) ... (28) 【0104】 【数29】 Pv = (bx, by, bz) ... (29) 【0105】なる式(28)、(29)で表される。 【0106】したがって、偏微分Pu及びPvは、式(25)～(27)の連立方程式に式(13)～(15)と、式(19)～(21)とを代入することにより算出することができる。 【0107】すなわち、このステップS1では、ステップS3で用いる偏微分Pu及びPvを、 【0108】 【数30】  <math display="block">a x = \frac{(x3-x1)*(y2-v1)-(x2-x1)*(y3-v1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (30)</math>   <math display="block">a y = \frac{(y3-y1)*(v2-v1)-(y2-y1)*(v3-v1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (31)</math>   <math display="block">a z = \frac{(z3-z1)*(v2-v1)-(z2-z1)*(v3-v1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (32)</math>   <math display="block">b x = \frac{(x2-x1)*(u3-u1)-(x3-x1)*(u2-u1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (33)</math>   <math display="block">b y = \frac{(y2-y1)*(u3-u1)-(y3-y1)*(u2-u1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (34)</math>   <math display="block">b z = \frac{(z2-z1)*(u3-u1)-(z3-z1)*(u2-u1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \dots (35)</math> </p> <p>【0114】なる式(30)～(35)により算出する。 【0115】また、ステップS1では、式(30)～(35)により得られた偏微分Pu及びPvを用いて、ステップS3で用いるパラメータGu及びGvを算出する。</p>	<p>10</p> <p>20</p> <p>30</p> <p>50</p>



(8)

特開平9-231402

13

14

【0116】ここで、パラメータGu及びGvを算出する際、物体の大きさに依存するバンプマッピングを行う場合には、  
\* 【0117】  
【数36】

$$G_u = \frac{P_u}{|P_u|^2} \quad \dots (36)$$

【0118】 ※ ※ 【数37】

$$G_v = \frac{P_v}{|P_v|^2} \quad \dots (37)$$

【0119】なる式(36)及び(37)を用いる。 ★ 【0121】  
【0120】また、物体の大きさに依存しないバンプマッピングを行う場合には、 ★

$$G_u = \frac{P_u}{|P_u|} \quad \dots (38)$$

【0122】 ☆ ☆ 【数39】

$$G_v = \frac{P_v}{|P_v|} \quad \dots (39)$$

【0123】なる式(38)及び(39)を用いる。 ◆ら得られる各頂点における視線ベクトルE1, E2, E3が。  
【0124】また、ステップS1では、ステップS2で用いるパラメータを求める際に、まず、ワールド座標系における視点の座標と、式(13)～(15)で与えられるワールド座標系における三角形の各頂点の座標とを◆

$$E_1 = (E_{x1}, E_{y1}, E_{z1}) \quad \dots (40)$$

【0126】 \* \* 【数41】

$$E_2 = (E_{x2}, E_{y2}, E_{z2}) \quad \dots (41)$$

【0127】 ※ ※ 【数42】

$$E_3 = (E_{x3}, E_{y3}, E_{z3}) \quad \dots (42)$$

【0128】なる式(40)～(42)、クリップ座標系における座標が(x, y, z, w)である点における(u, v)パラメータMが、 ★

$$M = (U, V) \quad \dots (43)$$

【0130】なる式(43)、法線ベクトルNが、 ☆ 【数44】

【0131】 ☆

$$N = (N_x, N_y, N_z) \quad \dots (44)$$

【0132】なる式(44)、視線ベクトルEが、 ◆ 【数45】

【0133】 ◆

$$E = (E_x, E_y, E_z) \quad \dots (45)$$

【0134】なる式(45)で与えられるものとする。 \*る。すなわち、"Ms", "Ns", 及び "Es" が、

【0135】また、(u, v)パラメータM、法線ベクトルN、及び視線ベクトルEを各々"w"で割ったものを "Ms", "Ns", 及び "Es" で表すものとす \*

$$\begin{aligned} M_s &= (S, T, Q) \\ &= (U/w, V/w, 1/w) \quad \dots (46) \end{aligned}$$

【0137】 ※ ※ 【数47】

$$\begin{aligned} E_s &= (E_{sx}, E_{sy}, E_{sz}) \\ &= (E_x/w, E_y/w, E_z/w) \quad \dots (47) \end{aligned}$$

【0138】 S0 【数48】

(9)

特開平9-231402

15

$$\begin{aligned} N_s &= (N_{sx}, N_{sy}, N_{sz}) \\ &= (N_x/w, N_y/w, N_z/w) \quad \dots (48) \end{aligned}$$

16

【0139】なる式(46)～(48)で与えられるものとする。

【0140】したがって、ステップS1では、ステップS2で用いるパラメータ、すなわち三角形の描画開始点における“Ms”、“Ns”、及び“Es”の各値と、“Ms”、“Ns”、及び“Es”の三角形の描画開始辺方向の増分と走査線方向の増分とを、式(16)～(24)、及び式(40)～(42)を用いて算出する。

【0141】尚、ステップS2で用いるパラメータの算出方法は、三角形の描画処理におけるカラー値やZ座標の値に対して行われる算出方法と同様であるため、その詳細な説明は省略する。

$$M_k = (S_k, T_k, Q_k) \quad \dots (49)$$

【0145】

※ ※【数50】

$$N_k = (N_{kx}, N_{ky}, N_{kz}) \quad \dots (50)$$

【0146】

★ ★【数51】

$$E_k = (E_{kx}, E_{ky}, E_{kz}) \quad \dots (51)$$

【0147】なる式(49)～(51)により算出する。

【0148】尚、線形縮間の処理については、三角形の描画処理におけるカラー値やZ座標の値に対して行われる処理と同様であるため、その詳細な説明は省略する。☆

$$(s, t) = (S_k/Q_k, T_k/Q_k) \quad \dots (52)$$

【0151】なる式(52)により算出する。

【0152】また、式(50)により得られた法線ベクトル $N_k$ を用いて、ステップS4における法線ベクトル $N^i$ の算出に必要なベクトル $n$ を、

【0153】

【数53】

$$\begin{aligned} n &= (n_x, n_y, n_z) \\ &= (N_{kx}/Q_k, N_{ky}/Q_k, N_{kz}/Q_k) \quad \dots (53) \end{aligned}$$

【0154】なる式(53)により算出する。

【0155】次に、ステップS3では、ステップS1で得られた偏微分 $P_u$ 、 $P_v$ 、及びパラメータ $G_u$ 、 $G_v$ と、ステップS2で得られたパラメータ $(s, t)$ を用いて、法線ベクトル $N$ の変化分 $D$ を求める。

【0156】ステップS3の処理について具体的に説明すると、まず、物体の大きさに依存するバンプマッピングでは、上述したように、法線ベクトル $N$ の変化分 $D$  ◆

$$N = P_u \times P_v$$

【0163】なる式(56)で与えられる。

【0164】これらの式(54)～(56)に示したようにして、近似を行うことにより、式(5)は、

\* 【0142】次に、ステップS2では、ステップS1で得られたパラメータを用いて、テクスチャ座標、法線ベクトル、及び視線ベクトルの縮間処理を行う。

【0143】ステップS2の処理について具体的に説明すると、まず、ステップS1で得られた三角形の描画開始点における“Ms”、“Ns”、及び“Es”の各値と、“Ms”、“Ns”、及び“Es”の三角形の描画開始辺方向の増分と走査線方向の増分とを用いて線形縮間することにより、三角形内部の各描画ピクセル毎のテクスチャ座標 $M_k$ 、法線ベクトル $N_k$ 、及び視線ベクトル $E_k$ を、

【0144】

【数49】

☆ 【0149】次に、式(49)により得られたテクスチャ座標 $M_k$ を用いて、ステップS3においてテクスチャを参照する際に必要なパラメータ $(s, t)$ を、

【0150】

【数52】

◆は、式(5)により求められる。

【0157】そこで、ここでは、偏微分 $P_u$ 及び $P_v$ は、互いに直交していると近似する。したがって、偏微分 $P_u$ 及び $P_v$ は、

【0158】

【数54】

$$(P_u \cdot P_v) \approx 0 \quad \dots (54)$$

【0159】

【数55】

$$|P_u \times P_v| \approx |P_u| * |P_v| \quad \dots (55)$$

【0160】なる式(54)及び(55)で与えられる。

【0161】また、法線ベクトル $N$ としては、三角形内で一定の値を使用する。したがって、法線ベクトル $N$ は、

【0162】

【数56】

$$\dots (56)$$

【0165】

【数57】

(10)

特開平9-231402

17

18

$$D = -|Pu| * |Pv| * (Fu * \frac{Pu}{|Pu|^2} + Fv * \frac{Pv}{|Pv|^2}) \dots (57)$$

【0166】なる式(57)に示すように、簡略化することができる。

【0167】一方、物体の大きさに依存しないバンプマッピングでは、上述したように、法線ベクトルNの変化分Dは、式(12)により求められる。

【0168】そこで、ここでは、物体の大きさに依存する\*

るバンプマッピングの場合と同様に、式(54)～(56)に示したようにして、近似することにより、式(12)は、  
【0169】  
【数58】

$$D = -|Pu| * |Pv| * (Fu * \frac{Pu}{|Pu|} + Fv * \frac{Pv}{|Pv|}) \dots (58)$$

【0170】なる式(58)に示すように、簡略化することができる。

【0171】ところで、式(57)及び(58)において、“Fu”と“Fv”は、ステップS2の式(52)で得られたパラメータ(s、t)を用いて、対象描画ピクセルに対応するテクスチャを参照することにより求め

る。  
※【0172】したがって、ステップS3では、上述のよう

うにして得られた“Fu”と“Fv”、及びステップS1の式(36)～(39)で得られた三角形内で一定のパラメータGu、Gvを用いて、ベクトルdを、  
【0173】  
【数59】

$$d = -(Fu * Gu + Fv * Gv) \dots (59)$$

【0174】なる式(59)により算出する。

【0175】次に、ステップS4では、ステップS2で得られた法線ベクトルNと、ステップS3で得られた法線ベクトルNの変化分Dを用いて、バンプが付いた面の法線ベクトルN\*を求める。

【0176】ステップS4の処理について具体的に説明★

★すると、まず、ステップS2で得られたベクトルnと、ステップS3で得られたベクトルdを用いた場合、バンプが付いた面の法線ベクトルN\*は、  
【0177】  
【数60】

$$N^* = |Pu| * |Pv| * (\frac{n}{|n|} + d) \dots (60)$$

【0178】なる式(60)に示すように、近似することができる。

【0179】ところで、式(60)において、ベクトルnの大きさが「1」であると近似する。これにより、式★

$$N^* = |Pu| * |Pv| * (n + d) \dots (61)$$

【0181】なる式(61)に示すように、さらに、簡略化することができる。

【0182】また、ステップS5のライティング計算においては、詳細は後述するが、法線ベクトルN\*の方向のみを使用し、ベクトルの大きさを無視することができ◆

◆る。

【0183】したがって、ステップS4では、法線ベクトルN\*を、  
【0184】  
【数62】

$$N^* = n + d \dots (62)$$

【0185】なる式(62)により算出する。

【0186】次に、ステップS5では、ステップS2で得られた視線ベクトルEkと、ステップS4で得られた法線ベクトルN\*を用いて、ライティング計算を行い、対象描画ピクセルの色を決定する。

【0187】ステップS5で行うライティング計算につ★

$$R^* = 2 * N^* \cdot (N^* \cdot V) - V \dots (63)$$

※いて具体的に説明すると、まず、一般的に、ライティング計算は、上述した式(10)により行われる。

【0188】ここで、視線に対する反射ベクトルR\*は、  
【0189】  
【数63】

【0190】なる式(63)で与えられ、式(63)に

50 おいて、“N”は法線ベクトルであり、“V”は視線ベ

(11)

特開平9-231402

19

20

クトルである。

【0191】また、図2に示すように、反射ベクトル $R^*$ とライトベクトル $L$ の成す角度( $=\alpha$ )は、視線ベクトル $V$ とライトに対する反射ベクトル $R$ の成す角度( $=\alpha$ )に等しい。

$$(R \cdot V) = (L \cdot R^*)$$

【0194】なる式(64)により表される。

【0195】この式(64)により、式(10)は、

$$I\lambda = I_a\lambda + k_a O_d\lambda + \sum f_{att} I_p\lambda [kd O_d\lambda (R \cdot L) + ks Os\lambda (R^* \cdot L)^*] \quad \dots (65)$$

【0197】なる式(65)に書き換えることができる。

【0198】ここで、ライトベクトル $L$ は、平行光源の場合には、光源毎に一定であるが、点光源の場合には、物体上の位置に依存する。

$$\sum f_{att} I_p\lambda (N \cdot L)$$

【0201】なる式(66)で表される拡散反射の項を求めるための係数(以下、拡散反射係数と言う。)と、

$$\sum f_{att} I_p\lambda (R^* \cdot L)^*$$

【0203】なる式(67)で表される鏡面反射の項を求めるための係数(以下、鏡面反射係数と言う。)とを予め算出する。そして、拡散反射係数用のテーブル(以下、拡散反射テーブルと言う。)、及び鏡面反射係数用のテーブル(以下、鏡面反射テーブルと言う。))に、式(66)及び(67)により得られた各係数を対応するテーブルに各々格納する。

【0204】以下、拡散反射テーブル及び鏡面反射テ

$$SIZE = 2^*$$

【0207】なる式(68)で与えられるものとする。

【0208】このとき、2次元配列 $[i, j]$ の要素と、立方体の各面のインデックス $l (= (lx, ly))$ 

$$lx = \frac{l}{2^{*1}} - 1$$

【0210】

$$ly = \frac{j}{2^{*1}} - 1$$

【0211】なる式(69)及び(70)で与えられる。

【0212】そこで、先ず、インデックス $l (= (lx, ly))$ で与えられる点と、立方体200の中心 $O$ を結ぶベクトルを正規化する。次に、正規化したベクトルを、式(66)及び(67)における視線ベクトル $N$ 及び反射ベクトル $R^*$ として、立方体200の各面に対する2次元配列 $[i, j]$ の全要素について算出する。そして、その算出結果を、拡散反射テーブル及び鏡面反射テーブルに各々格納する。

【0213】上述のようにして予め作成した拡散反射テ

\*【0192】したがって、視線ベクトル $V$ 、ライトに対する反射ベクトル $R$ 、反射ベクトル $R^*$ 、及びライトベクトル $L$ の関係は、

【0193】

\* 【数64】

... (64)

\* 【0196】

\* 【数65】

★【0199】そこで、点光源においては、描画点のワールド座標に関わらず、物体中心と光源を結ぶベクトルで近似し、式(65)において、

【0200】

★ 【数66】

... (66)

☆【0202】

\* 【数67】

... (67)

◆ブルの作成方法及び参照方法について、具体的に説明する。

【0205】まず、拡散反射テーブル及び鏡面反射テーブルは、図3に示すような立方体200の各面に対応する2次元配列 $[i, j]$ として定義する。また、各2次元配列 $[i, j]$ の幅と高さの各サイズ $SIZE$ が、

【0206】

◆ 【数68】

... (68)

30\*y)との対応は、

【0209】

\* 【数69】

... (69)

\* \* 【数70】

... (70)

40 ーブル及び鏡面反射テーブルを参照する際には、ベクトル方向の線分と、立方体200との交点を求め、求めた交点の座標値からインデックス $l (= (lx, ly))$ を求める。

【0214】すなわち、ベクトルのメジャー軸で立方体200の面を選択し、マイナー軸の値をメジャー軸の値で割ることにより、インデックス $l (= (lx, ly))$ を求める。

【0215】例えば、ベクトルが、

【0216】

50 【数71】

(12)

特開平9-231402

21

$$(X, Y, Z) = (-0.5, 3, 0, 2, 5) \dots (71)$$

22

【0217】なる式(71)により表される場合には、  
-X面を選択する。したがって、上記図3において、-  
X面を選択した場合、インデックスI (= (|X, |  
Y))は、

【0218】

【数72】

$$I = (|X, |Y) = \left(-\frac{Z}{X}, \frac{Y}{X}\right) \\ = (0.6, -0.5) \dots (72)$$

【0219】なる式(72)により算出される。

【0220】上述のように、予め作成した拡散反射テ-  
ブル及び鏡面反射テーブルを参照する際には、ベクトル  
の大きさは無関係であり、正規化する必要もない。

【0221】したがって、ステップS5では、ステップ  
S4で得られた正規化されていない法線ベクトルN\*を\*

$$A\lambda = |a\lambda| k\alpha Od\lambda \dots (74)$$

【0226】

$$Kd\lambda = kd Od\lambda \dots (75)$$

【0227】

$$Ks\lambda = ks Os\lambda \dots (76)$$

【0228】なる式(74)～(76)により表され、  
描画している物体に固有の値である。

【0229】次に、ステップS6では、ステップS2～  
ステップS5の各処理が全ての描画ピクセルに対して行  
われたか否かを判断し、全ての描画ピクセルへの処理が  
終了するまで、ステップS2～ステップS5の各処理を  
描画ピクセル毎に行う。そして、全ての描画ピクセルへ  
の処理が終了した場合に、次のステップS7の処理に進  
む。

【0230】ステップS7では、ステップS1～ステ-  
ップS5の各処理が描画する物体を構成する全ての三角形  
に対して行われたか否かを判断し、全ての三角形への処  
理が終了するまで、ステップS1～ステップS5の各処  
理を三角形毎に行う。そして、全ての三角形への処理が  
終了した場合に、本バンプマッピングの処理を終了とす  
る。

【0231】上述のように、上記図1に示したバンプマ  
ッピング方法では、描画する物体を三角形等の多角形の  
集合に近似すると共に、位置ベクトルの偏微分が多角  
形内部で一定であると近似するため、描画ピクセル毎に  
位置ベクトルを算出する必要はなく、1個の多角形に対  
して1回のみ、位置ベクトルの偏微分を算出すればよい。  
したがって、上記バンプマッピング方法は、バンプマ  
ッピングを行う際の計算量を削減することができる。

【0232】また、上記バンプマッピング方法では、バ  
ンプが付いた面の法線ベクトルを算出する際に必要なベ

\*用いて、拡散反射テーブルを参照することにより得られ  
た拡散反射係数Dcと、法線ベクトルN\*とステップS  
2で得られた正規化されていない視線ベクトルEkを式  
(63)に代入して得られた視線に対する反射ベクトル  
R\*を用いて、鏡面反射テーブルを参照することにより  
得られた鏡面反射係数Scとを用いて、対象描画ピクセ  
ルの色Iλを、

$$I\lambda = A\lambda + Kd\lambda * Dc\lambda + Ks\lambda * Sc\lambda \dots (73)$$

【0223】なる式(73)により算出する。

【0224】ここで、式(73)において、“Aλ”、  
“Kdλ”、及び“Ksλ”は、各々、

【0225】

【数74】

$$\dots (74)$$

※ ※ 【数75】

$$\dots (75)$$

★ ★ 【数76】

$$\dots (76)$$

クトルの算出を描画ピクセル毎に線形補間と式(53)  
により行い、テクスチャを参照する際に必要なパラメ-  
タの算出を式(52)により行うが、式(52)及び  
(53)では、“1/Q”を求めているため、式(5  
3)の算出処理は、3回の乗算で行うことができる。ま  
た、式(53)により得られたベクトルを用いて上記法  
線ベクトルを算出する際に、位置ベクトルの偏微分が多  
角形内部で一定であると近似し、多角形の各頂点におけ  
る“u”及び“v”の各値と座標値から上記法線ベクト  
ルを算出するため、描画ピクセル毎に位置ベクトルの偏  
微分を算出する必要はなく、1個の多角形に対して1回  
のみ位置ベクトルの偏微分を算出すればよい。したがっ  
て、上記バンプマッピング方法は、バンプマッピングを  
行う際の計算量を削減することができる。

【0233】また、上記バンプマッピング方法では、変  
換前の法線ベクトルを多角形の各頂点における法線ベク  
トルの線形補間により算出し、法線ベクトルの変化分を  
求める際には、位置ベクトルの偏微分が互いに直交して  
いると近似し、法線ベクトルとして式(56)で与えら  
れる多角形内で一定の値を使用しているため、法線ベク  
トルの変化分の算出を式(59)に示したような簡略化  
した式で行うことができる。すなわち、法線ベクトル  
の変化分の算出処理は、従来のバンプマッピング方法で  
は、式(5)に示したように、8回の加算処理、3回の  
減算処理、21回の乗算処理、1回の除算処理、及び1  
回のルート計算処理が必要であるが、上記図1に示した

(13)

特開平9-231402

23

バンプマッピング方法では、式(59)に示したように、3回の加算処理、及び6回の乗算処理のみで行うことができる。したがって、上記バンプマッピング方法は、画質を落とすことなく、バンプマッピングを行う際の計算量を大幅に削減することができる。

【0234】また、上記バンプマッピング方法では、ライティング計算処理において、式(66)の算出処理を予め行うことにより作成した立方体の各面に対応する2次元配列として定義した拡散反射テーブルと、式(67)の算出処理を予め行うことにより作成した立方体の各面に対応する2次元配列として定義した鏡面反射テーブルとを用いて、バンプが付いた面の法線ベクトルで拡散反射テーブルを参照し、視線に対する反射ベクトルで鏡面反射テーブルを参照することにより、対象描画ピクセルのカラー値を算出するため、ベクトルの正規化やべき乗の計算処理を行う必要がない。したがって、上記バンプマッピング方法は、バンプマッピングを行う際の計算量を削減することができる。

【0235】上述のことにより、上記図1に示したバンプマッピング方法は、従来と比べ、バンプマッピングの処理を行う際の計算量を大幅に削減することができるため、バンプマッピングを小規模のハードウェアで容易に実現することができる。したがって、上記バンプマッピング方法は、小規模のハードウェアで、画質を落とすことなく、高速にバンプマッピングの処理を行うことができる。

【0236】つぎに、本発明に係る画像生成装置について説明する。

【0237】例えば、図1に示すように、上記画像生成装置100は、レジスタ7と、レジスタ7の出力が供給される補間処理回路1と、補間処理回路1の出力が各々供給される除算器2及び反射ベクトル算出回路4と、除算器2及びレジスタ7の各出力が供給される法線算出回路3と、反射ベクトル算出回路4及び法線算出回路3の各出力が供給されるカラー算出回路5とを備えている。

【0238】また、画像生成装置100は、法線算出回路3及びカラー算出回路5と各々接続されたメモリインターフェース6と、メモリインターフェース6と接続されたメモリ8とを備えている。

【0239】さらに、除算器2とカラー算出回路5は、接続された構成としており、出力端子1...からは、カラー算出回路5の出力が供給されるようになされている。

【0240】まず、画像生成装置100は、上記図1に示したフローチャートに従ってバンプマッピングの処理を行うようになされている。

【0241】すなわち、画像生成装置100は、例えば、2つの変数 $u$ 、 $v$ を用いて定義されるパッチに分割された物体を描画する際、まず、図示していない前処理回路により、上記パッチを多角形の集合で近似する。こ

24

のとき、多角形の各頂点のモデリング座標系における座標値と法線ベクトル、及び $(u, v)$ 座標を頂点データとする。

【0242】次に、画像生成装置100は、上記前処理回路により、上記頂点データを多角形単位に座標変換してプリミティブデータを生成する。そして、画像生成装置100は、上記図4に示した各構成要素により、上記前処理回路で生成されたプリミティブデータに対してバンプマッピングの処理を行う。

【0243】以下、画像生成装置100におけるバンプマッピングの処理時の動作について、上記図1及び上記図4を用いて具体的に説明する。

【0244】画像生成装置100において、例えば、三角形のプリミティブデータに対してバンプマッピングの処理を行う場合、まず、レジスタ7には、上述のようにして、上記前処理回路により生成された三角形のプリミティブデータから求められた補間処理回路1及び法線算出回路3で用いられる三角形毎のパラメータが格納される。

【0245】すなわち、レジスタ7には、ステップS1の式(30)～(35)により、三角形のプリミティブデータに対して一意に算出された偏微分 $P_u$ 及び $P_v$ を用いて、物体の大きさに依存するバンプマッピングを行う場合には、式(36)及び(37)により、物体の大きさに依存しないバンプマッピングを行う場合には、式(38)及び(39)により算出されたパラメータ $G_u$ 及び $G_v$ が格納される。

【0246】また、レジスタ7には、ステップS1の式(16)～(24)、及び式(40)～(42)により算出された三角形の描画開始点における $M_s$ 、 $N_s$ 、及び $E_s$ の各値と、 $M_s$ 、 $N_s$ 、及び $E_s$ の三角形の描画開始辺方向の増分と走査線方向の増分とが格納される。

【0247】一方、メモリ8には、法線算出回路3で用いられるテクスチャ、及びカラー算出回路5で用いられる上述のようにして予め作成された拡散反射テーブルと鏡面反射テーブルが格納される。

【0248】補間処理回路1は、レジスタ7に格納された三角形の描画開始点における $M_s$ 、 $N_s$ 、及び $E_s$ の各値と、 $M_s$ 、 $N_s$ 、及び $E_s$ の三角形の描画開始辺方向の増分と走査線方向の増分とを用いて、線形補間を行い、ステップS1の式(49)～(51)で定義される三角形内部の各描画ピクセル毎のテクスチャ座標 $M_k$ 、法線ベクトル $N_k$ 、及び視線ベクトル $E_k$ を求める。

【0249】除算器2は、補間処理回路1で得られた各描画ピクセル毎のテクスチャ座標 $M_k$ 、法線ベクトル $N_k$ 、及び視線ベクトル $E_k$ のうちの、対象描画ピクセルに対するテクスチャ座標 $M_k$ 、法線ベクトル $N_k$ 、及び視線ベクトル $E_k$ を用いて、ステップS1の式(52)

(14)

特開平9-231402

25

26

及び(53)の算出処理を行うことにより、テクスチャを参照する際に必要なパラメータ $(s, t)$ 、及び法線ベクトル $N^*$ の算出に必要なベクトル $n_c$ を求める。

【0250】法線算出回路3は、除算器2で得られた対象描画ピクセルに対するパラメータ $(s, t)$ から、テクスチャに対する2次元アドレス $(u, v)$ を求める。

【0251】メモリインターフェース6は、法線算出回路3で得られた2次元アドレス $(u, v)$ を、メモリ8に格納されたテクスチャに対する物理アドレス $addr$ に変換し、変換して得られた物理アドレス $addr$ を用いて、メモリ8の物理アドレス $addr$ に格納されている対象描画ピクセルに対するステップS3で示した“Fu”及び“Fu”を信号線 $data$ を介して読み出し、読み出した“Fu”及び“Fu”を読出データTDとして法線算出回路3に供給する。

【0252】法線算出回路3は、メモリインターフェース6からの読出データTD、すなわち対象描画ピクセルに対する“Fu”及び“Fu”と、レジスタ7に格納されているパラメータ $G_u$ 及び $G_v$ と、除算器2で得られた対象描画ピクセルに対するベクトル $n$ を用いて、ステップS3の式(59)及びステップS4で示した式(62)の算出処理を行うことにより、パンプが付いた面の法線ベクトル $N^*$ を求める。

【0253】反射ベクトル算出回路4は、縮小処理回路1で得られた対象描画ピクセルに対する法線ベクトル $N^*$ 及び視線ベクトル $E_k$ を用いて、ステップS5で示した式(63)の算出処理を行うことにより、対象描画ピクセルの視線に対する反射ベクトル $R^*$ を求める。

【0254】カラー算出回路5は、法線算出回路3で得られた法線ベクトル $N^*$ において、メジャー軸の値を $MAJ$ 、マイナー軸の値を $MIN1$ 及び $MIN2$ として、除算器2に供給する。

【0255】除算器2は、ステップS5で述べたように、カラー算出回路5からのマイナー軸の値 $MIN1$ 及び $MIN2$ を、カラー算出回路5からのメジャー軸の値 $MAJ$ で割る算出処理を行うことにより、対象描画ピクセルの拡散テーブルに対するインデックス $I (= (I_x, I_y))$ を求める。そして、除算器2は、求めたインデックス $I (= (I_x, I_y))$ をカラー算出回路5に供給する。

【0256】カラー算出回路5は、除算器2からのインデックス $I (= (I_x, I_y))$ を用いて、ステップS5の式(69)及び(70)の算出処理を行うことにより、拡散反射テーブルに対する2次元アドレス $(i, j)$ を求め、求めた2次元アドレス $(i, j)$ をメモリインターフェース6に供給する。また、カラー算出回路5は、メジャー軸により決定された立方体の面を示す信号 $p1n$ 、及び拡散反射テーブルに対するアクセスであるか、又は鏡面反射テーブルに対するアクセスであるかを示す信号 $ds$ もメモリインターフェース6に供給す

る。

【0257】メモリインターフェース6は、カラー算出回路5からの2次元アドレス $(i, j)$ 、信号 $p1n$ 、及び信号 $ds$ から、メモリ8に格納された拡散反射テーブルに対する物理アドレスを求め、その物理アドレスに格納されているステップS5の式(73)における拡散反射係数 $D_c$ を信号線 $data$ を介して読み出す。そして、メモリインターフェース6は、メモリ8から読み出した拡散反射係数 $D_c$ を読出データTD<sup>\*</sup>としてカラー算出回路5に供給する。

【0258】また、メモリインターフェース6は、拡散反射係数 $D_c$ の読出処理と同様にして、カラー算出回路5からの2次元アドレス $(i, j)$ 、信号 $p1n$ 、及び信号 $ds$ から、メモリ8に格納された鏡面反射テーブルに対する物理アドレスを求め、その物理アドレスに格納されているステップS5の式(73)における鏡面反射係数 $S_c$ を信号線 $data$ を介して読み出す。そして、メモリインターフェース6は、メモリ8から読み出した鏡面反射係数 $S_c$ を読出データTD<sup>\*</sup>としてカラー算出回路5に供給する。

【0259】カラー算出回路5は、メモリインターフェース6から順次供給された読出データTD<sup>\*</sup>、すなわち拡散反射係数 $D_c$ 、及び鏡面反射係数 $S_c$ を用いて、ステップS5の式(73)の算出処理を行うことにより、対象描画ピクセルの色を決定し、決定した色に対応したカラー信号 $color$ を出力端子1...を介して出力する。

【0260】上述のように、画像生成装置100は、上記図1に示したフローチャートに従って、パンプマッピングを行うため、上記パンプマッピングを小規模なハードウェアで高速にパンプマッピングを行うことができると共に、良好な画質を得ることができる。

【0261】尚、上述した画像生成装置100では、パラメータ $(s, t)$ 、ベクトル $n_c$ 、及びインデックス $I (= (I_x, I_y))$ を除算器2を用いて求めることとしたが、除算器2を近似多項式を用いた加減算器に置き換えて、パラメータ $(s, t)$ 、ベクトル $n_c$ 、及びインデックス $I (= (I_x, I_y))$ を求めることとしてもよい。これにより、画像生成装置100は、さらに小規模のハードウェアで高速にパンプマッピングを行うことができる。

【0262】また、画像生成装置100では、テクスチャ、拡散反射テーブル、及び鏡面反射テーブルを同一のメモリ8に格納することとしたが、3つのメモリを設け、テクスチャ、拡散反射テーブル、及び鏡面反射テーブルを対応したメモリにそれぞれ格納することとしてもよい。これにより、画像生成装置100は、メモリアクセスを並列的に行うことができるため、処理性能をさらに向上させることができる。

【0263】

(15)

特開平9-231402

27

【発明の効果】本発明に係るバンブマッピング方法では、まず、2つの変数 $u$ 及び $v$ を用いて定義されるパッチを多角形の集合で近似する。次に、上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出する。次に、上記偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する。次に、テクスチャ関数を用いて上記法線ベクトルの変化分を算出する。そして、上記法線ベクトルと上記法線ベクトルの変化分を用いて、対象描画ピクセルのカラー値を算出するための変換後法線ベクトルを算出する。これにより、上記バンブマッピング方法では、描画ピクセル毎に、位置ベクトルを算出する必要はない。このため、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができるため、バンブマッピングを小規模のハードウェアで容易に実現することができる。したがって、上記バンブマッピング方法は、小規模のハードウェアでバンブマッピングを高速に実行することができる。

【0264】また、本発明に係るバンブマッピング方法では、上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数 $u$ 及び $v$ の値と座標値から上記偏微分を算出する。これにより、上記バンブマッピング方法では、描画ピクセル毎に、位置ベクトルを算出する必要はなく、1個の多角形に対して1回のみ、位置ベクトルの偏微分を算出すればよい。したがって、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができる。

【0265】また、本発明に係るバンブマッピング方法では、上記法線ベクトルを多角形の各頂点における法線ベクトルから算出する。これにより、描画ピクセル毎に、位置ベクトルを算出する必要はなく、1個の多角形に対して1回のみ、位置ベクトルの偏微分を算出すればよい。したがって、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができる。

【0266】また、本発明に係るバンブマッピング方法では、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出する。これにより、上記バンブマッピング方法は、法線ベクトルの変化分の算出を簡略化した式により行うことができる。したがって、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができる。

【0267】また、本発明に係るバンブマッピング方法では、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出する。これにより、上記バンブマッピング方法は、法線ベクトルの変化分の算出を簡略化した式により行うことができる。したがって、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができる。

28

【0268】また、本発明に係るバンブマッピング方法では、ライティング計算における拡散反射に関わる項を予め算出し、その算出結果を拡散反射テーブルに格納する。また、ライティング計算における鏡面反射に関わる項を予め算出し、その算出結果を鏡面反射テーブルに格納する。そして、上記変換後法線ベクトルで上記拡散反射テーブルを参照し、視線に対する反射ベクトルで上記鏡面反射テーブルを参照して対象描画ピクセルのカラー値を算出する。これにより、上記バンブマッピング方法では、ベクトルの正規化べき乗の計算が不要となる。したがって、上記バンブマッピング方法は、バンブマッピングを行う際の計算量を削減することができる。

【0269】本発明に係る画像生成装置は、偏微分算出手段は、2つの変数 $u$ 及び $v$ を用いて定義されるパッチを多角形の集合で近似して、上記パッチの変数 $u$ 及び $v$ 各々に対する偏微分を算出する。法線算出手段は、上記偏微分算出手段で得られた偏微分を用いて対象描画ピクセルに対する法線ベクトルを算出する。変化分算出手段は、上記法線算出手段で得られた法線ベクトルの変化分をテクスチャ関数を用いて算出する。変換後法線算出手段は、上記法線算出手段で得られた法線ベクトルと上記変化分算出手段で得られた法線ベクトルの変化分を用いて、対象描画ピクセルのカラー値を算出するための変換後法線ベクトルを算出する。これにより、上記画像生成装置では、描画ピクセル毎に、位置ベクトルを算出する必要はない。このため、上記画像生成装置は、バンブマッピングを行う際の計算量を削減することができるため、バンブマッピングを小規模のハードウェアで容易に実現することができる。したがって、上記画像生成装置は、小規模のハードウェアでバンブマッピングを高速に実行することができる。

【0270】また、本発明に係る画像生成装置では、上記偏微分算出手段は、上記偏微分は多角形内部で一定であると近似して、多角形の各頂点における変数 $u$ 及び $v$ の値と座標値から上記偏微分を算出する。これにより、上記画像生成装置では、描画ピクセル毎に、位置ベクトルを算出する必要はなく、1個の多角形に対して1回のみ、位置ベクトルの偏微分を算出すればよい。したがって、上記画像生成装置は、バンブマッピングを行う際の計算量を削減することができる。

【0271】また、本発明に係る画像生成装置では、上記法線算出手段は、上記法線ベクトルを多角形の各頂点における法線ベクトルから算出する。これにより、上記画像生成装置では、描画ピクセル毎に、位置ベクトルを算出する必要はなく、1個の多角形に対して1回のみ、位置ベクトルの偏微分を算出すればよい。したがって、上記画像生成装置は、バンブマッピングを行う際の計算量を削減することができる。

【0272】また、本発明に係る画像生成装置では、上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記



(15)

特開平9-231402

29

変数 $v$ に対する偏微分が直交していると近似して、上記法線ベクトルの変化分を算出する。これにより、上記画像生成装置は、法線ベクトルの変化分の算出を簡略化した式により行うことができる。したがって、上記画像生成装置は、バンプマッピングを行う際の計算量を削減することができる。

【0273】また、本発明に係る画像生成装置では、上記変化分算出手段は、上記変数 $u$ に対する偏微分と上記変数 $v$ に対する偏微分の外積により得られる多角形内部で一定の法線ベクトルにより、上記法線ベクトルの変化分を算出する。これにより、上記画像生成装置は、法線ベクトルの変化分の算出を簡略化した式により行うことができる。したがって、上記画像生成装置は、バンプマッピングを行う際の計算量を削減することができる。

【0274】また、本発明に係る画像生成装置では、拡散反射テーブルには、予め算出されたライティング計算における拡散反射に関わる項が格納される。また、鏡面反射テーブルには、予め算出されたライティング計算における鏡面反射に関わる項が格納される。そして、カラー値算出手段は、上記変換後法線算出手段で得られた変換後法線ベクトルで上記拡散反射テーブルを参照し、視線に対する反射ベクトルで上記鏡面反射テーブルを参照し、

※して上記カラー値を算出する。これにより、上記画像生成装置では、ベクトルの正規化べき乗の計算が不要となる。したがって、上記画像生成装置は、バンプマッピングを行う際の計算量を削減することができる。

【図面の簡単な説明】

【図1】本発明に係るバンプマッピング方法を示すフローチャートである。

【図2】ライティング計算において、法線ベクトル、視線ベクトル、視線に対する反射ベクトル、ライトベクトル、及びライトに対する反射ベクトルの関係を説明するための図である。

【図3】拡散反射テーブル及び鏡面反射テーブルの作成処理を説明するための図である。

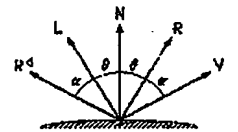
【図4】本発明に係る画像生成装置の構成を示すブロック図である。

【図5】従来のバンプマッピング方法を示すフローチャートである。

【符号の説明】

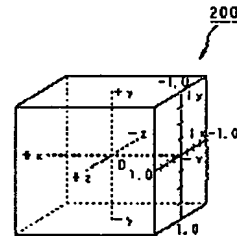
1 補間処理回路、2 除算器、3 法線算出回路、4 反射ベクトル算出回路、5 カラー算出回路、6 メモリインターフェース、7 レジスタ、8 メモリ、100 画像生成装置

【図2】



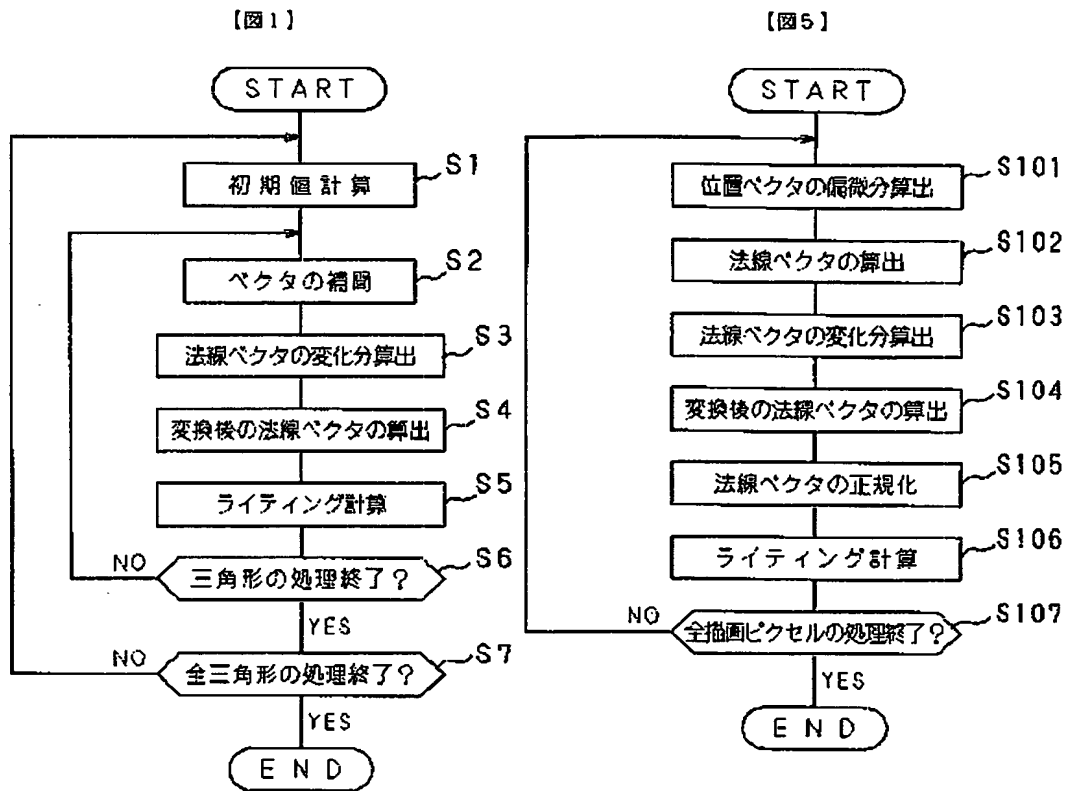
ライティング計算におけるベクトルの関係

【図3】



(17)

特開平9-231402



本発明に係るバンプマッピング方法

従来のバンプマッピング方法



**\* NOTICES \***

JPO and NCIPI are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**DETAILED DESCRIPTION**

---

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the image generation equipment which adopted the bump mapping approach for shading \*\*\*\* by the shading to express the small irregularity on a body, and the above-mentioned bump mapping approach in the computer graphics system which is the visual equipment which used the computer, special effect equipment, a video game machine, etc.

[0002]

[Description of the Prior Art] For example, a computer graphics system is a system which creates and displays an image and an image with a computer and a graphics peripheral device, and is utilized in many fields, such as a simulation of the reaction and response in the CAD system for the design exchange in a machine, the electrical and electric equipment, construction, etc., chemistry, aeronautical navigation, control, etc., education, art, and video game.

[0003] The system equipped with the image generation equipment which mainly utilizes the numerical calculation capacity of a computer and creates a three-dimensional image as a computer graphics system (It is hereafter called a three-dimension graphics system.) It is. This three-dimension graphics system It is the system which is performing frequently coordinate transformation called rotation, migration, expansion / contraction in the solid model in a calculating machine, and carries out a screen display, and is the system which needs advanced techniques, such as coordinate transformation, transparent transformation, shading processing, and a hidden line / hidden surface elimination processing, compared with a two-dimensional graphic system.

[0004] In recent years, the applicable field of the above three-dimension graphics systems is expanded rapidly, and is widely used also not only in CAD but in image work, video game, etc.

[0005] For this reason, it not only displays the configuration of an objective three dimension on a two-dimensional screen, but the capacity which generates an image with a more high reality is demanded of the image generation equipment with which a three-dimension graphics system is equipped.

[0006] Then, in order to obtain the image which raised the power of expression of graphics, the technique called texture mapping is adopted with the above-mentioned image generation equipment. This texture mapping is the technique of sticking the two-dimensional image independently prepared as a texture source image (pattern), i.e., a texture pattern, on the front face of fundamental unit graphic forms (polygon), such as a triangle which constitutes a body, and a square, and a thing equipped with the image generation equipment which performs texture mapping at a high speed by hardware is also in a three-dimension graphics system.

[0007] However, although texture mapping which was mentioned above could raise the power of expression about a color sharply, the image obtained by texture mapping had the problem that an objective front face was too smooth and missing at a reality.

[0008] Then, in order to solve this problem, there is image generation equipment which adopted the technique called the bump mapping into which texture mapping was developed. This bump mapping is the technique of expressing the heterogeneity and irregularity on a body front face in false, by mapping a normal vector and expressing the small irregularity on a body by shading \*\*\*\* by the shading.

[0009] Drawing 5 is a flow chart which shows processing of bump mapping. Hereafter, processing of bump mapping is explained using above-mentioned drawing 5.

[0010] For example, it is position vector P in the point P on the body corresponding to the pixel (henceforth a drawing pixel) which draws on a frame memory at step S101 first when the modeling of the body is carried out with the curve expressed with the polynomial of two variables u and v

[0011]

[Equation 1]

$$P = (X(u, v), Y(u, v), Z(u, v)) \quad \dots (1)$$

[0012] It computes by the becoming formula (1).

[0013] And they are the partial differentials Pu and Pv to "u" and "v" of position vector P which were obtained by the formula (1) [0014]

[Equation 2]

$$P_u = (\partial X / \partial u, \partial Y / \partial u, \partial Z / \partial u) \quad \dots (2)$$

[0015]

[Equation 3]

$$P_v = (\partial X / \partial v, \partial Y / \partial v, \partial Z / \partial v) \quad \dots (3)$$

[0016] It computes by the becoming formula (2) and (3).

[0017] Here, the partial differentials Pu and Pv obtained by a formula (2) and (3) exist on the tangential plane in the point P on a body.

[0018] So, at the following step S102, normal vector N in Point P is calculated by the outer product of the partial differentials Pu and Pv obtained by the formula (1). That is, at step S102, it is normal vector N [0019]

[Equation 4]

$$N = P_u \times P_v \quad \dots (4)$$

[0020] It computes by the becoming formula (4).

[0021] Next, at step S103, set to "F(u, v)" first the function which gives the height of the irregularity on an objective field (henceforth a bump), and let the partial differentials Fu and Fv to "u" and "v" of "F" be textures.

[0022] And with reference to the value of the texture which corresponds from each value of "u" in the point P on a body, and "v", the partial differentials Pu and Pv obtained at step S101 and normal vector N obtained at step S102 are used, and it is the change part D of normal vector N [0023]

[Equation 5]

$$D = \frac{F_u * (N \times P_v) - F_v * (N \times P_u)}{|N|} \quad \dots (5)$$

[0024] It asks by the becoming formula (5).

[0025] In addition, it sets at a ceremony (5) and normal vector N is  $N = (N_x, N_y, N_z)$

It comes out and "|N|" is [0026] at a certain time.

[Equation 6]

$$|N| = \sqrt{N_x^2 + N_y^2 + N_z^2} \quad \dots (6)$$

[0027] It is given by the becoming formula (6).

[0028] Next, it is the change part D of normal vector N obtained at step S102 in step S104, and normal vector N obtained at step S103 [0029]

[Equation 7]

$$N' = N + D \quad \dots (7)$$

[0030] As shown in the becoming formula (7), it asks for the normal vector Nd of the field where the

bump was attached by adding.

[0031] Next, it is the normal vector Nd obtained at step S104 in step S105 [0032]

[Equation 8]

$$Nn = \frac{Nd}{|Nd|} \quad \dots (8)$$

[0033] It asks for the normalization normal vector Nn by normalizing by the becoming formula (8).

[0034] In addition, it sets at a ceremony (8) and a normal vector Nd is Nd= (Ndx, Ndy, Ndz).

It comes out and "|Nd|" is [0035] at a certain time.

[Equation 9]

$$|Nd| = \sqrt{N^d x^2 + N^d y^2 + N^d z^2} \quad \dots (9)$$

[0036] It is given by the becoming formula (9).

[0037] Next, at step S106, using the normalization normal vector Nn obtained at step S105, lighting count is performed and the color of an object drawing pixel is determined.

[0038] At step S106, lighting count Namely, the color Ia of ambient light (Ambient Light), The color Ip of lighting, the attenuation factor fatt of a position light (Position Light) The environmental reflection coefficient ka (Ambient-reflection Coefficient), The diffuse reflection multiplier kd (Diffuse-reflection Coefficient), A specular reflection factor ks (Specular-reflection Coefficient), The objective diffusion color (Diffuse Color) Od, the objective specular color Os (Specular Color), It has exponent (Specular-reflection Exponent) n of normal vector N, the look vector V, the reflective vector R over a light, the light vector L, and a specular, and is [0039].

[Equation 10]

$$\lambda = Ia \lambda + \sum fatt \quad Ip \lambda [kd Od \lambda (N \cdot L) + ks Os \lambda (R \cdot V)^n] \quad \dots (10)$$

[0040] The color of an object drawing pixel is determined by carrying out by the becoming formula (10).

[0041] Here, in a formula (10), it is shown that "lambda" has a value for every color component.

[0042] Moreover, the reflective vector R over a light is [0043].

[Equation 11]

$$R = 2 * N * (N \cdot L) - L \quad \dots (11)$$

[0044] It is given by the becoming formula (11).

[0045] Moreover, in a formula (10) and a formula (11), normal vector N, the look vector V, the reflective vector R over a light, and the light vector L shall be normalized respectively.

[0046] In addition, about the lighting count mentioned above, since it is indicated by "Computer Graphics principles and practice (work besides Foly and VanDam)", the reference of "ADDISON WESLEY", etc., the detailed explanation is omitted.

[0047] Next, at step S107, it judges whether each processing of step S101 - step S106 was performed to all drawing pixels. And each processing of step S101 - step S106 is performed for every drawing pixel until the processing to all drawing pixels is completed.

[0048] By the way, in the image obtained by processing of bump mapping as shown in above-mentioned drawing 5, how whose false irregularity on the front face of a body is visible is dependent on the magnitude of the body to which bump mapping is performed. That is, in processing of the above-mentioned bump mapping, if the body to which bump mapping is performed is enlarged, flattening of the irregularity on the front face of a body is carried out.

[0049] However, it may be inconvenient if how whose false irregularity on the front face of a body is visible depending on the applicable field of a three-dimension graphics system is dependent on objective magnitude.

[0050] Then, in such a case, it sets at step S103, and is a formula (5) [0051]

[Equation 12]

$$D = \sqrt{F u^2 + F v^2} * |N| * \frac{Fu*(N \times Pv) - Fv*(N \times Pu)}{|Fu*(N \times Pv) - Fv*(N \times Pu)|} \dots (12)$$

[0052] It transposes to the becoming formula (12) and asks for the change part D of normal vector N. Thereby, the bump mapping independent of objective magnitude is realizable.

[0053] In addition, about the bump mapping mentioned above, since it is indicated by "SIMULATION OF WRINCHED SURFACES (James F.Blinn work)", the paper of "Proceed of SIGGRAPH'78", etc., the detailed explanation is omitted.

[0054]

[Problem(s) to be Solved by the Invention] However, by the conventional bump mapping approach which was mentioned above, as step S101 described, while asking for the position vector corresponding to a drawing pixel for every drawing pixel, it needed to ask for the partial differential of the above-mentioned position vector for every drawing pixel. For this reason, by the above-mentioned bump mapping approach, the computational complexity at the time of processing bump mapping was large.

[0055] Moreover, by the above-mentioned bump mapping approach, as step S102 described, it needed to ask for the normal vector for every drawing pixel using the partial differential of the position vector obtained at step S101. For this reason, by the above-mentioned bump mapping approach, the computational complexity at the time of processing bump mapping was large.

[0056] Moreover, as shown in the formula (5) or formula (12) of step S103, in case a changed part of the normal vector for every drawing pixel was computed, count of the root, the division, the outer product of a vector, etc. needed to be calculated of the above-mentioned bump mapping approach. For this reason, by the above-mentioned bump mapping approach, the computational complexity at the time of processing bump mapping was large.

[0057] Moreover, as step S106 described, in case lighting count for every drawing pixel was performed, as shown in the formula (10), count of a exponentiation was also required [ the normal vector the look vector, the light vector, and the reflective vector over a light needed to be normalized, and ] of the above-mentioned bump mapping approach. For this reason, by the above-mentioned bump mapping approach, the computational complexity at the time of processing bump mapping was large.

[0058] Therefore, since a hardware scale became very large when it is going to hardware-ize the above bump mapping, it was difficult to realize bump mapping by hardware. That is, the conventional image generation equipment which adopted bump mapping was not able to perform bump mapping by hardware.

[0059] Then, this invention is made in view of the conventional actual condition like \*\*\*\*, and has the following purposes.

[0060] That is, the purpose of this invention is to offer the bump mapping approach that bump mapping can be performed at a high speed by small-scale hardware.

[0061] Moreover, the purpose of this invention is to offer the image generation equipment which performs bump mapping at a high speed by small-scale hardware.

[0062]

[Means for Solving the Problem] In case the bump-mapping approach which starts this invention in order to solve an above-mentioned technical problem draws the patch defined using two variables u and v, it is the bump-mapping approach which computes the color value of an object drawing pixel using the normal vector after conversion which changed the normal vector to an object drawing pixel with the texture function, and was changed and obtained, and approximates the above-mentioned patch by polygonal set first. Next, the variable u of the above-mentioned patch and the partial differential to v of each are computed. Next, the normal vector to an object drawing pixel is computed using the above-mentioned partial differential. Next, a changed part of the above-mentioned normal vector is computed using a texture function. And it is characterized by computing the above-mentioned normal vector after conversion using a changed part of the above-mentioned normal vector and the above-mentioned normal vector.

[0063] Moreover, the bump mapping approach concerning this invention is characterized by for the

above-mentioned partial differential approximating that it is fixed inside a polygon, and computing the above-mentioned partial differential from the value and coordinate value of the variables  $u$  and  $v$  in each polygonal top-most vertices.

[0064] Moreover, the bump mapping approach concerning this invention is characterized by computing the above-mentioned normal vector from the normal vector in each polygonal top-most vertices.

[0065] Moreover, the bump mapping approach concerning this invention is characterized by computing a changed part of the above-mentioned normal vector by approximating, if the partial differential to the above-mentioned variable  $u$  and the partial differential to the above-mentioned variable  $v$  lie at right angles.

[0066] Moreover, the bump mapping approach concerning this invention is characterized by computing a changed part of the above-mentioned normal vector by the fixed normal vector inside the polygon obtained by the outer product of the partial differential to the above-mentioned variable  $u$ , and the partial differential to the above-mentioned variable  $v$ .

[0067] Moreover, the bump mapping approach concerning this invention computes beforehand the term in connection with the diffuse reflection in lighting count, and stores the calculation result in a diffuse reflection table. Moreover, the term in connection with the specular reflection in lighting count is computed beforehand, and the calculation result is stored in a specular reflection table. And it is characterized by computing the above-mentioned color value with reference to the above-mentioned specular reflection table by the reflective vector over a look with reference to the above-mentioned diffuse reflection table by the above-mentioned normal vector after conversion.

[0068] In order to solve an above-mentioned technical problem, the image generation equipment concerning this invention In case the patch defined using two variables  $u$  and  $v$  is drawn, the normal vector to an object drawing pixel is changed with a texture function. It is image generation equipment which adopted the bump mapping approach which computes the color value of an object drawing pixel using the normal vector after conversion changed and obtained. A partial-differential calculation means to compute the variable  $u$  of the above-mentioned patch, and the partial differential to  $v$  of each by approximating the above-mentioned patch by polygonal set, A normal calculation means to compute the normal vector to an object drawing pixel using the partial differential obtained with the above-mentioned partial-differential calculation means, A change part calculation means to compute a changed part of the normal vector obtained with the above-mentioned normal calculation means using a texture function, It is characterized by having a normal calculation means after conversion to compute the above-mentioned normal vector after conversion using a changed part of the normal vector obtained with the above-mentioned normal calculation means, and the normal vector obtained with the above-mentioned change part calculation means.

[0069] Moreover, the image generation equipment concerning this invention is characterized by approximating that the above-mentioned partial differential of the above-mentioned partial-differential calculation means is fixed inside a polygon, and computing the above-mentioned partial differential from the value and coordinate value of the variables  $u$  and  $v$  in each polygonal top-most vertices.

[0070] Moreover, the image generation equipment concerning this invention is characterized by the above-mentioned normal calculation means computing the above-mentioned normal vector from the normal vector in each polygonal top-most vertices.

[0071] Moreover, the image generation equipment concerning this invention is characterized by the above-mentioned change part calculation means computing a changed part of the above-mentioned normal vector by approximating it, if the partial differential to the above-mentioned variable  $u$  and the partial differential to the above-mentioned variable  $v$  lie at right angles.

[0072] Moreover, the image generation equipment concerning this invention is characterized by computing a changed part of the above-mentioned normal vector by the fixed normal vector inside the polygon from which the above-mentioned change part calculation means is acquired by the outer product of the partial differential to the above-mentioned variable  $u$ , and the partial differential to the above-mentioned variable  $v$ .

[0073] Moreover, the diffuse reflection table on which the term in connection with the diffuse reflection in the lighting count by which the image generation equipment concerning this invention



was computed beforehand is stored, The specular reflection table on which the term in connection with the specular reflection in the lighting count computed beforehand is stored, It is characterized by having a color value calculation means to compute the above-mentioned color value with reference to the above-mentioned specular reflection table by the reflective vector over a look with reference to the above-mentioned diffuse reflection table by the normal vector after conversion obtained with the above-mentioned normal calculation means after conversion.

[0074]

[Embodiment of the Invention] Hereafter, the gestalt of implementation of invention is explained to a detail with reference to a drawing.

[0075] First, the bump mapping approach concerning this invention is explained.

[0076] For example, in case the body divided into the patch defined using two variables  $u$  and  $v$  is drawn, the above-mentioned patch is first approximated by polygonal set. Let the coordinate value in the modeling system of coordinates of each polygonal top-most vertices, a normal vector, and  $(u, v)$  a coordinate be top-most-vertices data at this time.

[0077] Next, primitive data are generated by carrying out coordinate transformation of the above-mentioned top-most-vertices data in polygon. And bump mapping is processed to the generated primitive data.

[0078] Drawing 1 is a flow chart which shows processing of the above-mentioned bump mapping, and processing of the above-mentioned bump mapping applies the bump mapping approach concerning this invention.

[0079] Hereafter, processing of the above-mentioned bump mapping is concretely explained using above-mentioned drawing 1.

[0080] For example, when processing bump mapping to triangular primitive data, at step S1, it asks for a parameter required for interpolation processing of the vector in step S2, and changed calculation processing of the normal vector in step S3 first using the above-mentioned primitive data.

[0081] When processing of step S1 is explained concretely, the coordinates [ in / first / the world coordinate of each triangular top-most vertices ]  $P_1$ ,  $P_2$ , and  $P_3$  are [0082].

[Equation 13]

$$P_1 = (X_1, Y_1, Z_1) \quad \dots (13)$$

[0083]

[Equation 14]

$$P_2 = (X_2, Y_2, Z_2) \quad \dots (14)$$

[0084]

[Equation 15]

$$P_3 = (X_3, Y_3, Z_3) \quad \dots (15)$$

[0085] Becoming formula (13) The normal vectors  $N_1$ ,  $N_2$ , and  $N_3$  in - (15) and a world coordinate are [0086].

[Equation 16]

$$N_1 = (N_{x1}, N_{y1}, N_{z1}) \quad \dots (16)$$

[0087]

[Equation 17]

$$N_2 = (N_{x2}, N_{y2}, N_{z2}) \quad \dots (17)$$

[0088]

[Equation 18]

$$N_3 = (N_{x3}, N_{y3}, N_{z3}) \quad \dots (18)$$

[0089] Becoming formula (16) The parameters  $T_1$  and  $T_2$  of - (18), and  $(u, v)$  and  $T_3$  are [0090].

[Equation 19]

$$T1 = (u1, v1) \quad \dots (19)$$

[0091]

$$T2 = (u2, v2) \quad \dots (20)$$

[0092]

[Equation 21]

$$T3 = (u3, v3) \quad \dots (21)$$

[0093] Becoming formula (19) The coordinates Pc1, Pc2, and Pc3 in - (21) and clip system of coordinates are [0094].

[Equation 22]

$$Pc1 = (x1, y1, z1, w1) \quad \dots (22)$$

[0095]

[Equation 23]

$$Pc2 = (x2, y2, z2, w2) \quad \dots (23)$$

[0096]

[Equation 24]

$$Pc3 = (x3, y3, z3, w3) \quad \dots (24)$$

[0097] Becoming formula (22) It is respectively given by - (24) and the partial differentials Pu and Pv to "u" and "v" of a position vector approximate that it is fixed inside a triangle. At this time, the point on a triangular flat surface (X, Y, Z) is [0098].

[Equation 25]

$$X = ax * u + bx * v + cx \quad \dots (25)$$

[0099]

[Equation 26]

$$Y = ay * u + by * v + cy \quad \dots (26)$$

[0100]

[Equation 27]

$$Z = az * u + bz * v + cz \quad \dots (27)$$

[0101] Becoming formula (25) It is expressed with - (27).

[0102] Moreover, the partial differentials Pu and Pv to "u" and "v" of a position vector are [0103].

[Equation 28]

$$Pu = (ax, ay, az) \quad \dots (28)$$

[0104]

[Equation 29]

$$Pv = (bx, by, bz) \quad \dots (29)$$

[0105] It is expressed with the becoming formula (28) and (29).

[0106] Therefore, partial differentials Pu and Pv are computable by substituting equation (13) - (15) and equation (19) - (21) for the simultaneous equations of equation (25) - (27).

[0107] That is, they are the partial differentials Pu and Pv used at step S3 in this step S1 [0108]

[Equation 30]

$$ax = \frac{(x3-x1)*(v2-v1)-(x2-x1)*(v3-v1)}{(u3-u1)*(v2-v1)-(u2-u1)*(v3-v1)} \quad \dots (30)$$

[0109]

[Equation 31]

$$a_y = \frac{(y_3 - y_1) * (v_2 - v_1) - (y_2 - y_1) * (v_3 - v_1)}{(u_3 - u_1) * (v_2 - v_1) - (u_2 - u_1) * (v_3 - v_1)} \quad \dots (31)$$

[0110]

[Equation 32]

$$a_z = \frac{(z_3 - z_1) * (v_2 - v_1) - (z_2 - z_1) * (v_3 - v_1)}{(u_3 - u_1) * (v_2 - v_1) - (u_2 - u_1) * (v_3 - v_1)} \quad \dots (32)$$

[0111]

[Equation 33]

$$b_x = \frac{(x_2 - x_1) * (u_3 - u_1) - (x_3 - x_1) * (u_2 - u_1)}{(u_3 - u_1) * (v_2 - v_1) - (u_2 - u_1) * (v_3 - v_1)} \quad \dots (33)$$

[0112]

[Equation 34]

$$b_y = \frac{(y_2 - y_1) * (u_3 - u_1) - (y_3 - y_1) * (u_2 - u_1)}{(u_3 - u_1) * (v_2 - v_1) - (u_2 - u_1) * (v_3 - v_1)} \quad \dots (34)$$

[0113]

[Equation 35]

$$b_z = \frac{(z_2 - z_1) * (u_3 - u_1) - (z_3 - z_1) * (u_2 - u_1)}{(u_3 - u_1) * (v_2 - v_1) - (u_2 - u_1) * (v_3 - v_1)} \quad \dots (35)$$

[0114] Becoming formula (30) It computes by - (35).

[0115] Moreover, at step S1, the parameters Gu and Gv used at step S3 are computed using the partial differentials Pu and Pv obtained by formula (30) - (35).

[0116] It is [0117] when performing bump mapping depending on objective magnitude here, in case Parameters Gu and Gv are computed.

[Equation 36]

$$G_u = \frac{P_u}{|P_u|^2} \quad \dots (36)$$

[0118]

[Equation 37]

$$G_v = \frac{P_v}{|P_v|^2} \quad \dots (37)$$

[0119] The becoming formula (36) and (37) are used.

[0120] Moreover, it is [0121] when performing bump mapping independent of objective magnitude.

[Equation 38]

$$G_u = \frac{P_u}{|P_u|} \quad \dots (38)$$

[0122]

[Equation 39]

$$G_v = \frac{P_v}{|P_v|} \quad \dots (39)$$

[0123] The becoming formula (38) and (39) are used.

[0124] Moreover, the look vectors E1, E2, and E3 in each top-most vertices first obtained from the

coordinate of the view in a world coordinate and the coordinate of each top-most vertices of the triangle in the world coordinate given by formula (13) - (15) in case it asks for the parameter used at step S2 at step S1 are [0125].

$$\begin{aligned} & \text{[Equation 40]} \\ E_1 &= (E_{x1}, E_{y1}, E_{z1}) \quad \dots (40) \end{aligned}$$

[0126]

$$\begin{aligned} & \text{[Equation 41]} \\ E_2 &= (E_{x2}, E_{y2}, E_{z2}) \quad \dots (41) \end{aligned}$$

[0127]

$$\begin{aligned} & \text{[Equation 42]} \\ E_3 &= (E_{x3}, E_{y3}, E_{z3}) \quad \dots (42) \end{aligned}$$

[0128] Becoming formula (40) The parameter M in the point that the coordinates in - (42) and clip system of coordinates are x, and (y, z, w) (u, v) is [0129].

$$\begin{aligned} & \text{[Equation 43]} \\ M &= (U, V) \quad \dots (43) \end{aligned}$$

[0130] The becoming formula (43) and normal vector N are [0131].

$$\begin{aligned} & \text{[Equation 44]} \\ N &= (N_x, N_y, N_z) \quad \dots (44) \end{aligned}$$

[0132] The becoming formula (44) and the look vector E are [0133].

$$\begin{aligned} & \text{[Equation 45]} \\ E &= (E_x, E_y, E_z) \quad \dots (45) \end{aligned}$$

[0134] It shall be given by the becoming formula (45).

[0135] Moreover, Parameter (u, v) M, normal vector N, and the thing that broke the look vector E by "w" respectively shall be expressed with "Ms", "Ns", and "Es." That is, "Ms", "Ns", and "Es" are [0136].

$$\begin{aligned} & \text{[Equation 46]} \\ M_s &= (S, T, Q) \\ &= (U/w, V/w, I/w) \quad \dots (46) \end{aligned}$$

[0137]

$$\begin{aligned} & \text{[Equation 47]} \\ E_s &= (E_{sx}, E_{sy}, E_{sz}) \\ &= (E_x/w, E_y/w, E_z/w) \quad \dots (47) \end{aligned}$$

[0138]

$$\begin{aligned} & \text{[Equation 48]} \\ N_s &= (N_{sx}, N_{sy}, N_{sz}) \\ &= (N_x/w, N_y/w, N_z/w) \quad \dots (48) \end{aligned}$$

[0139] Becoming formula (46) It shall be given by - (48).

[0140] Therefore, at step S1, each value of "Ms" in the parameter used at step S2, i.e., a triangular drawing start point, "Ns", and "Es", and the increment of the direction of the drawing initiation side of the triangle of "Ms", "Ns", and "Es" and the increment of the direction of the scanning line are computed using formula (16) - (24) and type (40) - (42).

[0141] In addition, since the calculation approach of the parameter used at step S2 is the same as the calculation approach performed to the color value in triangular drawing processing, or the value of a Z coordinate, the detailed explanation is omitted.

[0142] Next, at step S2, interpolation processing of a texture coordinate, a normal vector, and a look vector is performed using the parameter obtained at step S1.

[0143] Each value of "Ms" in the drawing start point of the triangle first obtained at step S1 when processing of step S2 was explained concretely, "Ns", and "Es", It is the texture coordinate Mk, the normal vector Nk, and the look vector Ek for every drawing pixel inside a triangle by carrying out

linear interpolation using the increment of the direction of the drawing initiation side of the triangle of "Ms", "Ns", and "Es", and the increment of the direction of the scanning line [0144]

$$\text{[Equation 49]} \\ M_k = (S_k, T_k, Q_k) \quad \dots (49)$$

[0145]

$$\text{[Equation 50]} \\ N_k = (N_{kx}, N_{ky}, N_{kz}) \quad \dots (50)$$

[0146]

$$\text{[Equation 51]} \\ E_k = (E_{kx}, E_{ky}, E_{kz}) \quad \dots (51)$$

[0147] Becoming formula (49) It computes by - (51).

[0148] In addition, since it is the same as that of the processing performed to the color value in triangular drawing processing, or the value of a Z coordinate about processing of linear interpolation, the detailed explanation is omitted.

[0149] Next, it is a parameter (s, t) required in case a texture is referred to in step S3 using the texture coordinate  $M_k$  acquired by the formula (49) [0150]

$$\text{[Equation 52]} \\ (s, t) = (S_k / Q_k, T_k / Q_k) \quad \dots (52)$$

[0151] It computes by the becoming formula (52).

[0152] Moreover, it is the vector  $n$  required for calculation of the normal vector  $N_d$  in step S4 using the normal vector  $N_k$  obtained by the formula (50) [0153]

$$\text{[Equation 53]} \\ n = (n_x, n_y, n_z) \\ = (N_{kx} / Q_k, N_{ky} / Q_k, N_{kz} / Q_k) \quad \dots (53)$$

[0154] It computes by the becoming formula (53).

[0155] Next, at step S3, it asks for the change part  $D$  of normal vector  $N$  using the partial differentials  $P_u$  and  $P_v$  obtained at step S1 and Parameters  $G_u$  and  $G_v$ , and the parameter (s, t) obtained at step S2.

[0156] If processing of step S3 is explained concretely, as mentioned above, by the bump mapping depending on objective magnitude, the change part  $D$  of normal vector  $N$  will be first called for by the formula (5).

[0157] So, partial differentials  $P_u$  and  $P_v$  are approximated here, if it intersects perpendicularly mutually and is. Therefore, partial differentials  $P_u$  and  $P_v$  are [0158].

$$\text{[Equation 54]} \\ (P_u \cdot P_v) \simeq 0 \quad \dots (54)$$

[0159]

$$\text{[Equation 55]} \\ |P_u \times P_v| \simeq |P_u| * |P_v| \quad \dots (55)$$

[0160] It is given by the becoming formula (54) and (55).

[0161] Moreover, as normal vector  $N$ , a fixed value is used within a triangle. Therefore, normal vector  $N$  is [0162].

$$\text{[Equation 56]} \\ N = P_u \times P_v \quad \dots (56)$$

[0163] It is given by the becoming formula (56).

[0164] These formulas (54) A formula (5) is [0165] by approximating by [ as having been shown in - (56) ].

[Equation 57]

$$D = - |P_u| * |P_v| * \left( F_u * \frac{P_u}{|P_u|^2} + F_v * \frac{P_v}{|P_v|^2} \right) \quad \dots (57)$$

[0166] It can simplify, as shown in the becoming formula (57).

[0167] On the other hand, in the bump mapping independent of objective magnitude, as mentioned above, the change part D of normal vector N is called for by the formula (12).

[0168] So, a formula (12) is [0169] by resembling it, as it was shown in formula (54) - (56) like the case of the bump mapping depending on objective magnitude here.

[Equation 58]

$$D = - |P_u| * |P_v| * \left( F_u * \frac{P_u}{|P_u|} + F_v * \frac{P_v}{|P_v|} \right) \quad \dots (58)$$

[0170] It can simplify, as shown in the becoming formula (58).

[0171] Here, in a formula (57) and (58), "Fu" and "Fv" are calculated by referring to the texture corresponding to an object drawing pixel using the parameter (s, t) obtained by the formula (52) of step S2.

[0172] therefore -- a step -- S -- three -- \*\*\*\* -- above -- carrying out -- obtaining -- having had -- " - Fu -- " -- " -- Fu -- " -- and -- a step -- S -- one -- a formula -- (-- 36 --) - (-- 39 --) -- obtaining -- having had -- a triangle -- inside -- being fixed -- a parameter -- Gu -- Gv -- using -- Vector d --

[0173]

[Equation 59]

$$d = - (F_u * G_u + F_v * G_v) \quad \dots (59)$$

[0174] It computes by the becoming formula (59).

[0175] Next, in step S4, it asks for the normal vector Nd of the field where the bump was attached using the change part D of normal vector N obtained at step S2, and normal vector N obtained at step S3.

[0176] The normal vector Nd of the field where the bump was attached when processing of step S4 was explained concretely and the vector n acquired at step S2 and the vector d acquired at step S3 were used first is [0177].

[Equation 60]

$$N^d = |P_u| * |P_v| * \left( \frac{n}{|n|} + d \right) \quad \dots (60)$$

[0178] It can approximate, as shown in the becoming formula (60).

[0179] Here, in a formula (60), it approximates that the magnitude of Vector n is "1." Thereby, a formula (60) is [0180].

[Equation 61]

$$N^d = |P_u| * |P_v| * (n + d) \quad \dots (61)$$

[0181] As shown in the becoming formula (61), it can simplify further.

[0182] Moreover, in lighting count of step S5, although mentioned later for details, only the direction of a normal vector Nd can be used and the magnitude of a vector can be disregarded.

[0183] Therefore, at step S4, it is a normal vector Nd [0184]

[Equation 62]

$$N^d = n + d \quad \dots (62)$$

[0185] It computes by the becoming formula (62).

[0186] Next, at step S5, using the look vector Ek acquired at step S2, and the normal vector Nd obtained by step S4, lighting count is performed and the color of an object drawing pixel is determined.

[0187] If the lighting count performed at step S5 is explained concretely, generally lighting count will be first performed by the formula (10) mentioned above.

[0188] Here, the reflective vector Rd over a look is [0189].

[Equation 63]

$$R' = \frac{2 * N * (N \cdot V) - V}{\dots} \quad \dots (63)$$

[0190] It is given by the becoming formula (63), and in a formula (63), "N" is a normal vector and "V" is a look vector.

[0191] Moreover, as shown in drawing 2, the include angle (=alpha) which the reflective vector Rd and the light vector L accomplish is equal to the include angle (=alpha) which the look vector V and the reflective vector R over a light accomplish.

[0192] Therefore, the relation of the look vector V, the reflective vector R over a light, the reflective vector Rd, and the light vector L is [0193].

[Equation 64]

$$(R \cdot V) = (L \cdot R') \quad \dots (64)$$

[0194] It is expressed by the becoming formula 64.

[0195] By this formula (64), a formula (10) is [0196].

[Equation 65]

$$f_{\lambda} = I_a \lambda_{ka} O d \lambda + \sum f a t t_{i p \lambda} [k d O d \lambda (N \cdot L) + k s O s \lambda (R' \cdot L)^n] \quad \dots (65)$$

[0197] It can rewrite at a becoming ceremony (65).

[0198] Here, in the case of the parallel light source, for every light source, although the light vector L is fixed, in the case of the point light source, it depends for it in the location on a body.

[0199] Then, in the point light source, it is not concerned with the world coordinate of a drawing point, but it approximates by the vector which connects the light source to a body core, it sets at a ceremony (65), and is [0200].

[Equation 66]

$$\sum f a t t_{i p \lambda} (N \cdot L) \quad \dots (66)$$

[0201] The multiplier (henceforth a diffuse reflection multiplier) for searching for the term of the diffuse reflection expressed with the becoming formula (66), and [0202]

[Equation 67]

$$\sum f a t t_{i p \lambda} (R' \cdot L)^n \quad \dots (67)$$

[0203] The multiplier (henceforth a specular reflection factor) for searching for the term of the specular reflection expressed with the becoming formula (67) is computed beforehand. And each multiplier obtained by a formula (66) and (67) at the table for diffuse reflection multipliers (henceforth a diffuse reflection table) and the table for specular reflection factors (henceforth a specular reflection table) is respectively stored in a corresponding table.

[0204] Hereafter, the creation approach of a diffuse reflection table and a specular reflection table and the reference approach are explained concretely.

[0205] First, a diffuse reflection table and a specular reflection table are defined as two-dimensional array [i, j] corresponding to each side of the cube 200 as shown in drawing 3. Moreover, each size SIZE of the width of face and the height of a 2-dimensional each array [i, j] is [0206].

[Equation 68]

$$S I Z E = 2^n \quad \dots (68)$$

[0207] It shall be given by the becoming formula (68).

[0208] At this time, correspondence with the element of two-dimensional array [i, j] and the index I of each cubical side (= (Ix, Iy)) is [0209].

[Equation 69]

$$I x = \frac{i}{2^{n-1}} - 1 \quad \dots (69)$$

[0210]

[Equation 70]

$$I_y = \frac{j}{2^{n-1}} - 1 \quad \dots (70)$$

[0211] It is given by the becoming formula (69) and (70).

[0212] Then, the point given by Index I (= (Ix, Iy)) and the vector which connects the core O of a cube 200 are normalized first. Next, the vector which normalized is computed about all the elements of the two-dimensional array [i, j] to each side of a cube 200 as a formula (66), normal vector N in (67), and a reflective vector Rd. And the calculation result is respectively stored in a diffuse reflection table and a specular reflection table.

[0213] In case the diffuse reflection table and specular reflection table which were created beforehand as mentioned above are referred to, it asks for the segment of the direction of a vector, and an intersection with a cube 200, and Index I (= (Ix, Iy)) is calculated from the coordinate value of the intersection for which it asked.

[0214] That is, it asks for Index I (= (Ix, Iy)) by choosing the field of a cube 200 with the major shaft of a vector, and breaking the value of a minor shaft by the value of a major shaft.

[0215] For example, a vector is [0216].

$$\begin{aligned} & \text{[Equation 71]} \\ (X, Y, Z) &= (-0.5, 3.0, 2.5) \quad \dots (71) \end{aligned}$$

[0217] -X side is chosen when expressed by the becoming formula (71). Therefore, when -X side is chosen in above-mentioned drawing 3, Index I (= (Ix, Iy)) is [0218].

$$\begin{aligned} & \text{[Equation 72]} \\ I &= (I_x, I_y) = \left( -\frac{Z}{X}, \frac{Y}{X} \right) \\ &= (0.6, -0.5) \quad \dots (72) \end{aligned}$$

[0219] It is computed by the becoming formula (72).

[0220] As mentioned above, in case the diffuse reflection table and specular reflection table which were created beforehand are referred to, the magnitude of a vector is unrelated and it is not necessary to normalize it.

[0221] Therefore, the diffuse reflection multiplier Dc obtained by referring to a diffuse reflection table at step S5 using the normal vector Nd which was obtained by step S4, and which it does not normalize, The reflective vector Rd over the look which substituted for the formula (63) the normal vector Nd and the look vector Ek which was acquired at step S2, and which it does not normalize, and was obtained is used. The specular reflection factor Sc obtained by referring to a specular reflection table is used, and it is color Ilambda of an object drawing pixel [0222]

$$I_\lambda = A_\lambda + K_d \lambda * D_c \lambda + K_s \lambda * S_c \lambda \quad \dots (73)$$

[0223] It computes by the becoming formula (73).

[0224] Here, it sets at a ceremony (73) and "Alambda", "Kdlambda", and "Kslambda" are [0225] respectively.

$$A_\lambda = I_a \lambda \quad k_a \quad O_d \lambda \quad \dots (74)$$

[0226]

$$K_d \lambda = k_d \quad O_d \lambda \quad \dots (75)$$

[0227]

$$K_s \lambda = k_s \quad O_s \lambda \quad \dots (76) \text{ <BR>}$$

[0228] Becoming formula (74) It is the value of a proper at the body which was expressed by - (76)



and has drawn.

[0229] Next, at step S6, each processing of step S2 - step S5 is performed for every drawing pixel until it judges whether each processing of step S2 - step S5 was performed to all drawing pixels and the processing to all drawing pixels is completed. And when the processing to all drawing pixels is completed, it progresses to processing of the following step S7.

[0230] Each processing of step S1 - step S5 is performed for every triangle until it judges whether it was carried out to all the triangles that constitute the body which each processing of step S1 - step S5 draws from a step S7 and the processing to all triangles is completed. And when the processing to all triangles is completed, processing of this bump mapping is considered as termination.

[0231] As mentioned above, what is necessary is not to compute a position vector for every drawing pixel, and just to compute the partial differential of a position vector once to the polygon of one piece by the bump mapping approach shown in above-mentioned drawing 1, since it approximates that the partial differential of a position vector is fixed inside a polygon while approximating the body which draws to the set of polygons, such as a triangle. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0232] Moreover, in case the normal vector of the field where the bump was attached is computed, a required vector is computed by linear interpolation and the formula (53) for every drawing pixel, by the above-mentioned bump mapping approach, in case a texture is referred to, a required parameter is computed by the formula (52), but since it is asking for " $1/Q$ ", calculation processing of a formula (53) can be performed by 3 times of multiplication a formula (52) and (53). Moreover, in case the above-mentioned normal vector is computed using the vector acquired by the formula (53) In order to approximate that the partial differential of a position vector is fixed inside a polygon and to compute the above-mentioned normal vector from each value and coordinate value of " $u$ " in each polygonal top-most vertices, and " $v$ ", What is necessary is not to compute the partial differential of a position vector for every drawing pixel, and just to compute the partial differential of a position vector once to the polygon of one piece. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0233] Moreover, in case the normal vector before conversion is computed by the linear interpolation of the normal vector in each polygonal top-most vertices and it asks for a changed part of a normal vector by the above-mentioned bump mapping approach, it approximates, if the partial differential of a position vector lies at right angles mutually, and since the fixed value is used within the polygon given by the formula (56) as a normal vector, it can carry out by the simplified formula as showed changed calculation of a normal vector in the formula (59). That is, by the conventional bump mapping approach, as shown in the formula (5), eight addition processings, three subtraction processings, 21 multiplication processings, one division processing, and 1 time of root computation are required for changed calculation processing of a normal vector, but by the bump mapping approach shown in above-mentioned drawing 1, as shown in the formula (59), it can be performed only by three addition processings and six multiplication processings. Therefore, the above-mentioned bump mapping approach can reduce sharply the computational complexity at the time of performing bump mapping, without lowering image quality.

[0234] Moreover, by the above-mentioned bump mapping approach, it sets to lighting computation. The diffuse reflection table defined as two-dimensional array corresponding to each side of the cube created by performing calculation processing of a formula (66) beforehand, The specular reflection table defined as two-dimensional array corresponding to each side of the cube created by performing calculation processing of a formula (67) beforehand is used. In order to compute the color value of an object drawing pixel by referring to a specular reflection table by the reflective vector over a look with reference to a diffuse reflection table by the normal vector of the field where the bump was attached, it is necessary to perform neither normalization of a vector, nor computation of a exponentiation. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0235] By the above-mentioned thing, since the bump mapping approach shown in above-mentioned drawing 1 can reduce sharply the computational complexity at the time of processing bump mapping compared with the former, it can realize bump mapping easily by small-scale hardware. Therefore,

the above-mentioned bump mapping approach is small-scale hardware, and it can process bump mapping at a high speed, without lowering image quality.

[0236] Below, the image generation equipment concerning this invention is explained.

[0237] For example, as shown in drawing 1, the above-mentioned image generation equipment 100 has a register 7, the interpolation processing circuit 1 to which the output of a register 7 is supplied, the divider 2 and the reflective vector calculation circuit 4 to which the output of the interpolation processing circuit 1 is supplied respectively, the normal calculation circuit 3 to which each output of a divider 2 and a register 7 is supplied, and the color calculation circuit 5 to which each output of a reflective vector calculation circuit 4 and a normal calculation circuit 3 is supplied.

[0238] Moreover, image generation equipment 100 is equipped with the memory 8 connected with the memory interface 6 respectively connected with the normal calculation circuit 3 and the color calculation circuit 5, and the memory interface 6.

[0239] Furthermore, the divider 2 and the color calculation circuit 5 are considered as the connected configuration, and are made as [ supply / the output of the color calculation circuit 5 ] from the output terminal Iout.

[0240] First, image generation equipment 100 is made as [ process / according to the flow chart shown in above-mentioned drawing 1 / bump mapping ].

[0241] That is, in case image generation equipment 100 draws the body divided into the patch defined using two variables  $u$  and  $v$ , it approximates the above-mentioned patch by polygonal set by the pretreatment circuit which is not illustrated first. Let the coordinate value in the modeling system of coordinates of each polygonal top-most vertices, a normal vector, and  $(u, v)$  a coordinate be top-most-vertices data at this time.

[0242] Next, by the above-mentioned pretreatment circuit, image generation equipment 100 carries out coordinate transformation of the above-mentioned top-most-vertices data in polygon, and generates primitive data. And image generation equipment 100 processes bump mapping to the primitive data generated in the above-mentioned pretreatment circuit with each component shown in above-mentioned drawing 4.

[0243] Hereafter, the actuation at the time of processing of the bump mapping in image generation equipment 100 is concretely explained using above-mentioned drawing 1 and above-mentioned drawing 4.

[0244] In image generation equipment 100, when processing bump mapping to triangular primitive data, the parameter for every triangle used in the interpolation processing circuit 1 and the normal calculation circuit 3 which were called for from the primitive data of the triangle generated by the above-mentioned pretreatment circuit as mentioned above is first stored in a register 7.

[0245] To a register 7, namely, by formula [ of step S1 ] (30) - (35) In performing bump mapping depending on objective magnitude using the partial differentials  $P_u$  and  $P_v$  computed by the meaning to triangular primitive data When a formula (36) and (37) perform bump mapping independent of objective magnitude, the parameters  $G_u$  and  $G_v$  computed by a formula (38) and (39) are stored.

[0246] Moreover, each value of "Ms" in the drawing start point of the triangle computed by formula (16) - (24) and type (40) - (42) of step S1, "Ns", and "Es", and the increment of the direction of the drawing initiation side of the triangle of "Ms", "Ns", and "Es" and the increment of the direction of the scanning line are stored in a register 7.

[0247] On the other hand, the texture used in the normal calculation circuit 3 and the diffuse reflection table which is used in the color calculation circuit 5 and which was created beforehand as mentioned above, and a specular reflection table are stored in memory 8.

[0248] Each value of "Ms" in the triangular drawing start point by which the interpolation processing circuit 1 was stored in the register 7, "Ns", and "Es", The increment of the direction of the drawing initiation side of the triangle of "Ms", "Ns", and "Es" and the increment of the direction of the scanning line are used. Linear interpolation is performed and the texture coordinate  $M_k$ , the normal vector  $N_k$ , and the look vector  $E_k$  for every drawing pixel inside the triangle defined by formula [ of step S1 ] (49) - (51) are searched for.

[0249] The texture coordinate  $M_k$  for every drawing pixel from which the divider 2 was obtained in the interpolation processing circuit 1, The texture coordinate  $M_k$  over the object drawing pixel of a normal vector  $N_k$  and the look vectors  $E_k$ , Using a normal vector  $N_k$  and the look vector  $E_k$ , by

performing the formula (52) of step S1, and calculation processing of (53), in case a texture is referred to, a required parameter (s, t) and the vector  $nc$  required for calculation of a normal vector  $Nd$  are searched for.

[0250] The normal calculation circuit 3 asks for the two-dimensional address (u, v) to a texture from the parameter (s, t) to the object drawing pixel obtained with the divider 2.

[0251] The memory interface 6 the two-dimensional address (u, v) obtained in the normal calculation circuit 3 It changes into physical address  $addr$  to the texture stored in memory 8. "Fu" shown at step S3 to the object drawing pixel stored in physical address  $addr$  of memory 8 and "Fu" are read through a signal line data using physical address  $addr$  changed and obtained. Read "Fu" and "Fu" are supplied to the normal calculation circuit 3 as read-out data TD.

[0252] The normal calculation circuit 3 asks for the normal vector  $Nd$  of the field where the bump was attached by performing calculation processing of a formula (62) indicated to be "Fu" to the read-out data TD, i.e., the object drawing pixel, from the memory interface 6 and "Fu", and the parameters  $Gu$  and  $Gv$  stored in the register 7 by the formula (59) and step S4 of step S3 using the vector  $n$  over the object drawing pixel obtained with the divider 2.

[0253] The reflective vector calculation circuit 4 searches for the reflective vector  $Rd$  over the look of an object drawing pixel by performing calculation processing of a formula (63) shown at step S5 using the normal vector  $Nk$  to the object drawing pixel obtained in the interpolation processing circuit 1, and the look vector  $Ek$ .

[0254] In the normal vector  $Nd$  obtained in the normal calculation circuit 3, the color calculation circuit 5 sets the value of MAJ and a minor shaft to MIN1 and MIN2, and supplies the value of a major shaft to a divider 2.

[0255] A divider 2 asks for the index  $I$  to the diffusion table of an object drawing pixel ( $= (Ix, Iy)$ ) by performing calculation processing which breaks the values MIN1 and MIN2 of the minor shaft from the color calculation circuit 5 by the value MAJ of the major shaft from the color calculation circuit 5, as step S5 described. And a divider 2 supplies the index  $I$  ( $= (Ix, Iy)$ ) for which it asked to the color calculation circuit 5.

[0256] Using the index  $I$  from a divider 2 ( $= (Ix, Iy)$ ), by performing the formula (69) of step S5, and calculation processing of (70), the color calculation circuit 5 asks for the two-dimensional address (i, j) to a diffuse reflection table, and supplies the two-dimensional address (i, j) for which it asked to the memory interface 6. Moreover, the color calculation circuit 5 also supplies the signal  $ds$  which shows whether it is access to the signal  $pln$  which shows the field of the cube determined with the major shaft, and a diffuse reflection table, or it is access to a specular reflection table to the memory interface 6.

[0257] The memory interface 6 asks for the physical address to the diffuse reflection table stored in memory 8, and reads the diffuse reflection multiplier  $Dc$  in the formula (73) of step S5 stored in the physical address through a signal line data from the two-dimensional address (i, j) from the color calculation circuit 5, Signal  $pln$ , and Signal  $ds$ . And the memory interface 6 supplies the diffuse reflection multiplier  $Dc$  read from memory 8 to the color calculation circuit 5 as read-out data TDd.

[0258] Moreover, like read-out processing of the diffuse reflection multiplier  $Dc$ , the memory interface 6 asks for the physical address to the specular reflection table stored in memory 8, and reads the specular reflection factor  $Sc$  in the formula (73) of step S5 stored in the physical address through a signal line data from the two-dimensional address (i, j) from the color calculation circuit 5, Signal  $pln$ , and Signal  $ds$ . And the memory interface 6 supplies the specular reflection factor  $Sc$  read from memory 8 to the color calculation circuit 5 as read-out data TDd.

[0259] By performing calculation processing of the formula (73) of step S5 using the read-out data TDd  $Dc$  by which sequential supply was carried out, i.e., a diffuse reflection multiplier, and a specular reflection factor  $Sc$  from the memory interface 6, the color calculation circuit 5 determines the color of an object drawing pixel, and outputs color signal color corresponding to the determined color through an output terminal  $Iout$ .

[0260] As mentioned above, it can acquire good image quality while it can perform bump mapping for the above-mentioned bump mapping at a high speed by small-scale hardware, since image generation equipment 100 performs bump mapping according to the flow chart shown in above-mentioned drawing 1.

[0261] In addition, although [ the image generation equipment 100 mentioned above ] it asks for a parameter (s, t), Vector nc, and Index I (= (Ix, Iy)) using a divider 2, it is good also as transposing a divider 2 to the adder subtracter using an approximate polynomial, and asking for a parameter (s, t), Vector nc, and Index I (= (Ix, Iy)). Thereby, image generation equipment 100 can perform bump mapping at a high speed by still more nearly small-scale hardware.

[0262] Moreover, although [ image generation equipment 100 ] a texture, a diffuse reflection table, and a specular reflection table are stored in the same memory 8, it is good also as preparing three memory and storing respectively a texture, a diffuse reflection table, and a specular reflection table in the corresponding memory. Thereby, since image generation equipment 100 can perform memory access in juxtaposition, it can raise the processing engine performance further.

[0263]

[Effect of the Invention] By the bump mapping approach concerning this invention, the patch defined using two variables u and v is first approximated by polygonal set. Next, the variable u of the above-mentioned patch and the partial differential to v of each are computed. Next, the normal vector to an object drawing pixel is computed using the above-mentioned partial differential. Next, a changed part of the above-mentioned normal vector is computed using a texture function. And the normal vector after conversion for computing the color value of an object drawing pixel is computed using a changed part of the above-mentioned normal vector and the above-mentioned normal vector. Thereby, it is not necessary to compute a position vector for every drawing pixel by the above-mentioned bump mapping approach. For this reason, since the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping, it can realize bump mapping easily by small-scale hardware. Therefore, the above-mentioned bump mapping approach can perform bump mapping at a high speed by small-scale hardware.

[0264] Moreover, by the bump mapping approach concerning this invention, the above-mentioned partial differential approximates that it is fixed inside a polygon, and computes the above-mentioned partial differential from the value and coordinate value of the variables u and v in each polygonal top-most vertices. What is necessary is thereby, not to compute a position vector and just to compute the partial differential of a position vector once to the polygon of one piece by the above-mentioned bump mapping approach, for every drawing pixel. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0265] Moreover, by the bump mapping approach concerning this invention, the above-mentioned normal vector is computed from the normal vector in each polygonal top-most vertices. What is necessary is thereby, not to compute a position vector and just to compute the partial differential of a position vector once to the polygon of one piece for every drawing pixel. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0266] Moreover, by the bump mapping approach concerning this invention, a changed part of the above-mentioned normal vector is computed by approximating, if the partial differential to the above-mentioned variable u and the partial differential to the above-mentioned variable v lie at right angles. Thereby, the formula which simplified changed calculation of a normal vector can perform the above-mentioned bump mapping approach. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0267] Moreover, by the bump mapping approach concerning this invention, a changed part of the above-mentioned normal vector is computed by the fixed normal vector inside the polygon obtained by the outer product of the partial differential to the above-mentioned variable u, and the partial differential to the above-mentioned variable v. Thereby, the formula which simplified changed calculation of a normal vector can perform the above-mentioned bump mapping approach. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0268] Moreover, by the bump mapping approach concerning this invention, the term in connection with the diffuse reflection in lighting count is computed beforehand, and the calculation result is stored in a diffuse reflection table. Moreover, the term in connection with the specular reflection in lighting count is computed beforehand, and the calculation result is stored in a specular reflection table. And with reference to the above-mentioned diffuse reflection table, the color value of an

object drawing pixel is computed with reference to the above-mentioned specular reflection table by the reflective vector over a look by the above-mentioned normal vector after conversion. Thereby, by the above-mentioned bump mapping approach, count of the normalization exponentiation of a vector becomes unnecessary. Therefore, the above-mentioned bump mapping approach can reduce the computational complexity at the time of performing bump mapping.

[0269] The image generation equipment concerning this invention computes the variable  $u$  of the above-mentioned patch, and the partial differential to  $v$  of each by a partial-differential calculation means approximating the patch defined using two variables  $u$  and  $v$  by polygonal set. A normal calculation means computes the normal vector to an object drawing pixel using the partial differential obtained with the above-mentioned partial-differential calculation means. A change part calculation means computes a changed part of the normal vector obtained with the above-mentioned normal calculation means using a texture function. The normal calculation means after conversion computes the normal vector after conversion for computing the color value of an object drawing pixel using a changed part of the normal vector obtained with the above-mentioned normal calculation means, and the normal vector obtained with the above-mentioned change part calculation means. Thereby, it is not necessary to compute a position vector for every drawing pixel with the above-mentioned image generation equipment. For this reason, since the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping, it can realize bump mapping easily by small-scale hardware. Therefore, the above-mentioned image generation equipment can perform bump mapping at a high speed by small-scale hardware.

[0270] Moreover, with the image generation equipment concerning this invention, the above-mentioned partial differential approximates the above-mentioned partial-differential calculation means as it is fixed inside a polygon, and it computes the above-mentioned partial differential from the value and coordinate value of the variables  $u$  and  $v$  in each polygonal top-most vertices. What is necessary is thereby, not to compute a position vector and just to compute the partial differential of a position vector once to the polygon of one piece with the above-mentioned image generation equipment, for every drawing pixel. Therefore, the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping.

[0271] Moreover, with the image generation equipment concerning this invention, the above-mentioned normal calculation means computes the above-mentioned normal vector from the normal vector in each polygonal top-most vertices. What is necessary is thereby, not to compute a position vector and just to compute the partial differential of a position vector once to the polygon of one piece with the above-mentioned image generation equipment, for every drawing pixel. Therefore, the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping.

[0272] Moreover, with the image generation equipment concerning this invention, the above-mentioned change part calculation means computes a changed part of the above-mentioned normal vector by approximating it, if the partial differential to the above-mentioned variable  $u$  and the partial differential to the above-mentioned variable  $v$  lie at right angles. Thereby, the formula which simplified changed calculation of a normal vector can perform the above-mentioned image generation equipment. Therefore, the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping.

[0273] Moreover, with the image generation equipment concerning this invention, the above-mentioned change part calculation means computes a changed part of the above-mentioned normal vector by the fixed normal vector inside the polygon obtained by the outer product of the partial differential to the above-mentioned variable  $u$ , and the partial differential to the above-mentioned variable  $v$ . Thereby, the formula which simplified changed calculation of a normal vector can perform the above-mentioned image generation equipment. Therefore, the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping.

[0274] Moreover, with the image generation equipment concerning this invention, the term in connection with the diffuse reflection in the lighting count computed beforehand is stored in a diffuse reflection table. Moreover, the term in connection with the specular reflection in the lighting

count computed beforehand is stored in a specular reflection table. And a color value calculation means computes the above-mentioned color value with reference to the above-mentioned specular reflection table by the reflective vector over a look with reference to the above-mentioned diffuse reflection table by the normal vector after conversion obtained with the above-mentioned normal calculation means after conversion. Thereby, with the above-mentioned image generation equipment, count of the normalization exponentiation of a vector becomes unnecessary. Therefore, the above-mentioned image generation equipment can reduce the computational complexity at the time of performing bump mapping.

---

[Translation done.]

**\* NOTICES \***

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**DESCRIPTION OF DRAWINGS**

---

[Brief Description of the Drawings]

[Drawing 1] It is the flow chart which shows the bump mapping approach concerning this invention.

[Drawing 2] In lighting count, it is drawing for explaining the relation of a normal vector, a look vector, the reflective vector over a look, a light vector, and the reflective vector over a light.

[Drawing 3] It is drawing for explaining creation processing of a diffuse reflection table and a specular reflection table.

[Drawing 4] It is the block diagram showing the configuration of the image generation equipment concerning this invention.

[Drawing 5] It is the flow chart which shows the conventional bump mapping approach.

[Description of Notations]

1 Interpolation Processing Circuit, 2 Divider, 3 Normal Calculation Circuit, 4 Reflective Vector Calculation Circuit, 5 Color Calculation Circuit, 6 Memory Interface, 7 Register, 8 Memory, 100 Image Generation Equipment

---

[Translation done.]

\* NOTICES \*

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

CLAIMS

---

[Claim(s)]

[Claim 1] In case the patch defined using two variables  $u$  and  $v$  is drawn, the normal vector to an object drawing pixel is changed with a texture function. It is the bump mapping approach which computes the color value of an object drawing pixel using the normal vector after conversion changed and obtained. Approximate the above-mentioned patch by polygonal set, and the variable  $u$  of the above-mentioned patch and the partial differential to  $v$  of each are computed. The bump mapping approach characterized by computing the normal vector to an object drawing pixel using the above-mentioned partial differential, computing a changed part of the above-mentioned normal vector using a texture function, and computing the above-mentioned normal vector after conversion using a changed part of the above-mentioned normal vector and the above-mentioned normal vector.

[Claim 2] The above-mentioned partial differential is the bump mapping approach according to claim 1 characterized by approximating that it is fixed inside a polygon and computing the above-mentioned partial differential from the value and coordinate value of the variables  $u$  and  $v$  in each polygonal top-most vertices.

[Claim 3] The bump mapping approach according to claim 1 characterized by computing the above-mentioned normal vector from the normal vector in each polygonal top-most vertices.

[Claim 4] The bump mapping approach according to claim 1 characterized by computing a changed part of the above-mentioned normal vector by approximating if the partial differential to the above-mentioned variable  $u$  and the partial differential to the above-mentioned variable  $v$  lie at right angles.

[Claim 5] The bump mapping approach according to claim 1 characterized by computing a changed part of the above-mentioned normal vector by the fixed normal vector inside the polygon obtained by the outer product of the partial differential to the above-mentioned variable  $u$ , and the partial differential to the above-mentioned variable  $v$ .

[Claim 6] The bump-mapping approach according to claim 1 characterized by to compute beforehand the term in connection with the diffuse reflection in lighting count, to store the calculation result in a diffuse-reflection table, to compute beforehand the term in connection with the specular reflection in lighting count, to store the calculation result in a specular-reflection table, and to compute the above-mentioned color value with reference to the above-mentioned specular-reflection table by the reflective vector over a look with reference to the above-mentioned diffuse-reflection table by the above-mentioned normal vector after conversion.

[Claim 7] In case the patch defined using two variables  $u$  and  $v$  is drawn, the normal vector to an object drawing pixel is changed with a texture function. It is image generation equipment which adopted the bump mapping approach which computes the color value of an object drawing pixel using the normal vector after conversion changed and obtained. A partial-differential calculation means to compute the variable  $u$  of the above-mentioned patch, and the partial differential to  $v$  of each by approximating the above-mentioned patch by polygonal set, A normal calculation means to compute the normal vector to an object drawing pixel using the partial differential obtained with the above-mentioned partial-differential calculation means, A change part calculation means to compute a changed part of the normal vector obtained with the above-mentioned normal calculation means using a texture function, Image generation equipment characterized by having a normal calculation means after conversion to compute the above-mentioned normal vector after conversion using a



changed part of the normal vector obtained with the above-mentioned normal calculation means, and the normal vector obtained with the above-mentioned change part calculation means.

[Claim 8] For the above-mentioned partial differential, the above-mentioned partial-differential calculation means is image generation equipment according to claim 7 characterized by approximating that it is fixed inside a polygon and computing the above-mentioned partial differential from the value and coordinate value of the variables  $u$  and  $v$  in each polygonal top-most vertices.

[Claim 9] The above-mentioned normal calculation means is image generation equipment according to claim 7 characterized by computing the above-mentioned normal vector from the normal vector in each polygonal top-most vertices.

[Claim 10] The above-mentioned change part calculation means is image generation equipment according to claim 7 characterized by computing a changed part of the above-mentioned normal vector by approximating if the partial differential to the above-mentioned variable  $u$  and the partial differential to the above-mentioned variable  $v$  lie at right angles.

[Claim 11] The above-mentioned change part calculation means is image generation equipment according to claim 7 characterized by computing a changed part of the above-mentioned normal vector by the fixed normal vector inside the polygon obtained by the outer product of the partial differential to the above-mentioned variable  $u$ , and the partial differential to the above-mentioned variable  $v$ .

[Claim 12] The diffuse reflection table on which the term in connection with the diffuse reflection in the lighting count computed beforehand is stored, The specular reflection table on which the term in connection with the specular reflection in the lighting count computed beforehand is stored, Image generation equipment according to claim 7 characterized by having a color value calculation means to compute the above-mentioned color value with reference to the above-mentioned specular reflection table by the reflective vector over a look with reference to the above-mentioned diffuse reflection table by the normal vector after conversion obtained with the above-mentioned normal calculation means after conversion.

---

[Translation done.]

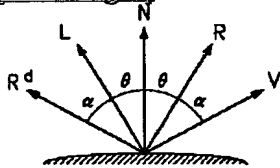
\* NOTICES \*

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

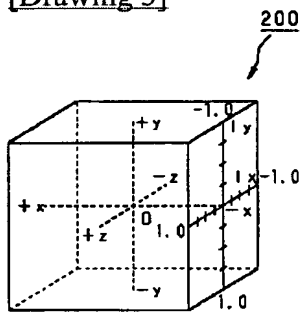
## DRAWINGS

[Drawing 2]

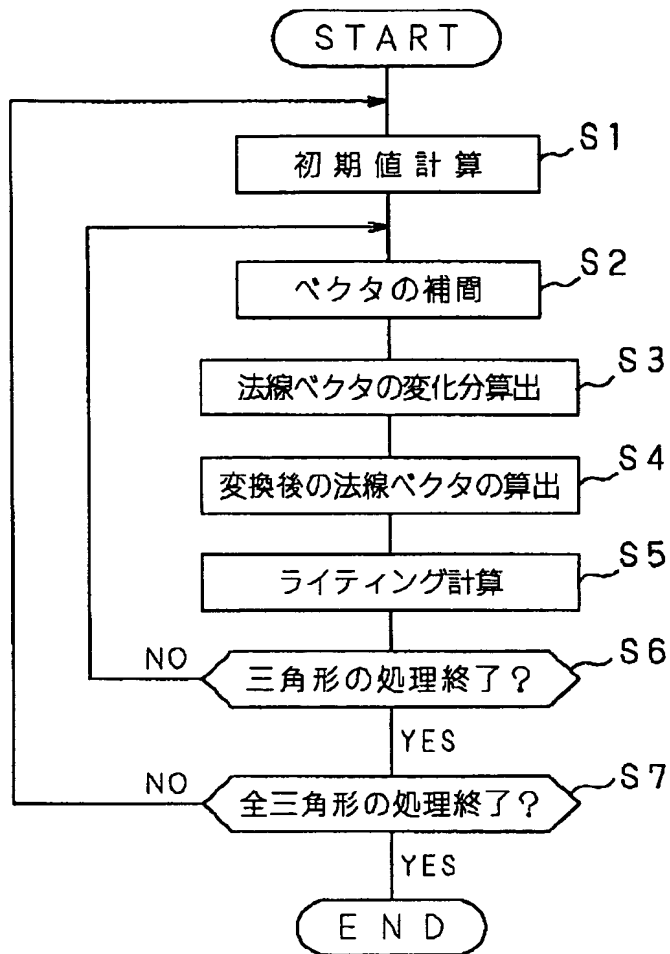


ライティング計算における各ベクトルの関係

[Drawing 3]

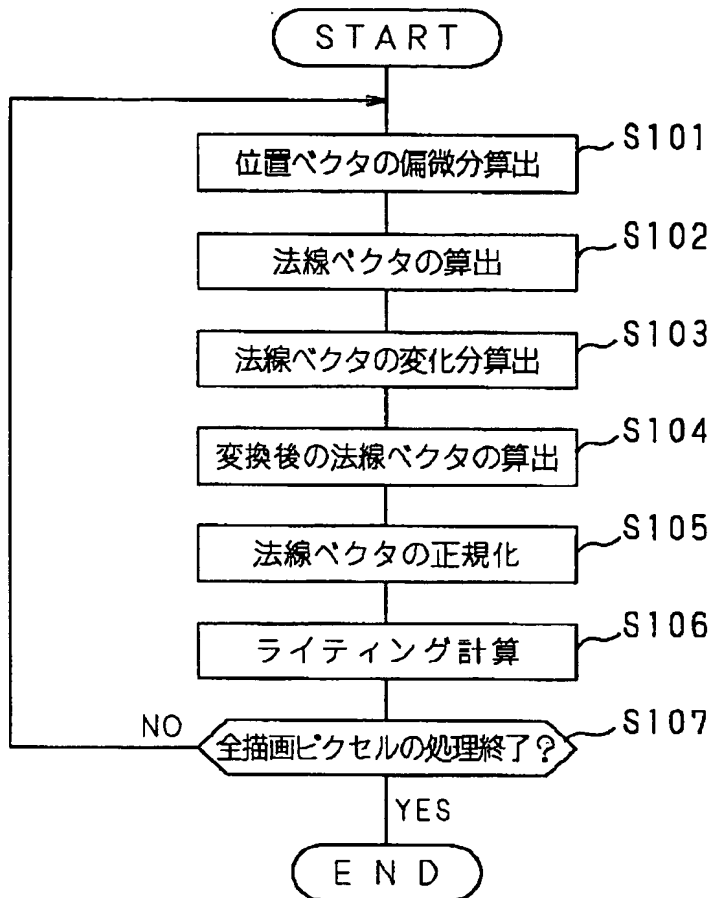


[Drawing 1]



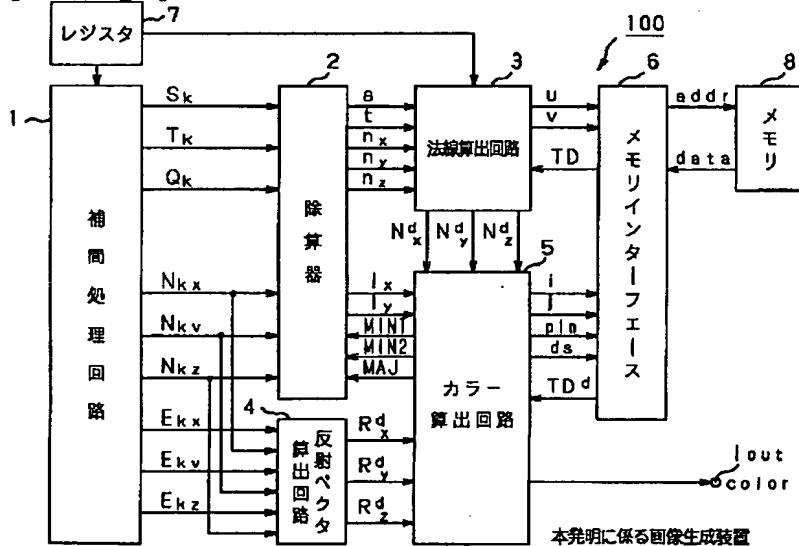
本発明に係るバンプマッピング方法

[Drawing 5]



従来のバンプマッピング方法

[Drawing 4]



[Translation done.]

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**